universität wien

DISSERTATION

Titel der Dissertation

„Statistical Challenges in Modern Genetics“

Verfasser

Muhammad FAISAL

angestrebter akademischer Grad

Doktor der Sozial- und Wirtschaftswissenschaften
(Dr. rer. soc. oec.)

Wien, im Jänner 2012,

Sudienkennzahl lt. Studienblatt:	A 084 136
Dissertationsgebiet lt. Studienblatt:	Statistik
Betreuer:	Ao. Univ.-Prof. Mag. Dr. Andreas Futschik

This page intentionally left blank

Dedicated

to

My Loving Parents

This page intentionally left blank

Abstract

In the past decade, remarkable advances have been made in the field of biology. Nowadays, biologists who study natural populations of plants and animals, have access to numerous new tools such as whole genome sequencing, DNA hybridization microarrays, and next-generation sequencing. Computationally intensive statistical methods have to be developed often for the analysis of complicated biological data. Of course, the advancement in the field of computing has been equally significant, and today's computers are fast enough to allow numerically intensive analysis to be run on desktop machines. This has led to a substantial progress in developing statistical methods for genetics; in particular, Markov chain Monte Carlo (MCMC) and Approximate Bayesian Computation (ABC) methods for computing likelihoods and posterior probabilities. The main objective of this study is to deal with statistical challenges in modern genetics. Both likelihood and likelihood-free methods are needed for the analysis of genetic data in the context of questions of interest to biologists. In this thesis, we contribute to both approaches. We propose a novel method for the estimation of time dependent scaled mutation rates under the infinite sites model when recombination is not present. The proposed method can also estimate time-independent mutation rates, and it performs well compared to other methods in the literature. Second, we investigate a method for choosing summary statistics to be used with the ABC algorithm. Our approach performs better in terms of computational time and accuracy than other methods given in the literature. Moreover, four new algorithms have been proposed for choosing the acceptance cutoff in ABC framework.

This page intentionally left blank

Declaration

The work in this thesis is based on research carried out at the Department of Statistics and Operations Research (ISOR), University of Vienna, Austria. No part of this thesis has been submitted somewhere else for any other degree or qualification, and it is all my own work, unless referenced, to the contrary, in the text.

Acknowledgments

All praise to Almighty ALLAH, the Benevolent, who bestowed upon me His blessings and through the mediation of His beloved Prophet Muhammad (PBUH) enlightened me with abundant resoluteness and perseverance which enable to me to accomplish this scientific assignment objectively and successfully.

Standing at the end of this long voyage of writing a dissertation, and looking back, I feel indebted to express my thanks to several people who at various stages of this thesis supported me in different ways.

I offer my humble and heartfelt gratitude to my laudable research advisor, **Prof. Andreas Futschik**, for introducing me to this fascinating area as well as his unswerving support and guidance. He always provided me with new ideas and feedback throughout my research work. Andreas statistical expertise and intuition as well as his focus on results had a significant impact on the outcome of this thesis. It was a great honour to experience his integrity, dedication and zeal for research and teaching. Further, I want to thank **Prof. Claus Vogl** for his enthusiastic participation in our joint work and for sharing his broad knowledge of topic.

I am obliged to the Higher Education Commission (HEC) of Pakistan for the grant of scholarship to pursue Ph.D. studies in Austria. I am grateful to the Austrian Exchange Service (OeAD) for regulating the scholarship installments.

There are so many people, whom I always wanted to thank for their helpful and caring attitude, yet I could not do so properly. I am still unable to mention the names of all those; but I hope they will understand that I never overlooked any of

them. Many thanks are due to my friends in Vienna for their cheerful company and assistance. It was due to them that I felt at home in Vienna. I owe to submit thanks to my friends and family members in Pakistan for their sincere wishes and prayers.

I am at a loss of words to express the gratitude towards my father, my mother, my brothers and my sisters. It was hard living far away from them; but their lifelong love, continuous encouragement, truthful prayers and belief in my abilities made me stronger, and I always felt their presence by my side.

Muhammad FAISAL

January 18, 2012

Preface

Statistical developments have in many cases been driven by applications in science. While genetics was always an important area that encouraged statistical research, recent technological advances in this discipline pose ever new and challenging problems to statisticians. This thesis covers some of the challenges arising from statistical inference in modern genetics. The key results from my thesis are covered in two submitted papers:

- Faisal M., Futschik A.,and Vogl C. Exact Likelihood Computation of Infinite Sites Model. Submitted to Theoretical Population Biology
- Faisal M. and Futschik A. Choosing Summary Statistics for Approximate Bayesian Computation. Submitted to Statistical Applications in Genetics and Molecular Biology

List of Abbreviations

AS	Asymptotic Sufficient
ABC	Approximate Bayesian Computation
ABC-MCMC	ABC with Markov Chain Monte Carlo
ABC-SMC	ABC with Sequential Monte Carlo
ABC-PRC	ABC with Partial Rejection Control
ABC-PMC	ABC with Population Monte Carlo
ABC-REJ	ABC with Rejection sampling
ABC-GLM	ABC with General Linear Model
EGT	Either-Griffiths-Tavaré
HGT	Horizontal Gene Transfer
LARS	Least Angle Regression
MRCA	Most Recent Common Ancestor
ME	Maximum Entropy
PLS	Partial Least Square
DNA	Deoxyribonucleic Acid
RNA	Ribonucleic Acid

Contents

List of Figures

List of Tables

This page intentionally left blank

Chapter 1

Introduction

1.1 Motivation

The last decade has witnessed a revolution in the field of Biology; nowadays, biologists have databases containing genomes of many organisms. The advent of high-throughput data collection methods in biotechnology, such as whole genome sequencing, *DNA* hybridization micro-arrays, and protein structure determination has created a large amount of data with incompletely understood information. There are a lot of data out there at a click of the button, and researchers have access to this tremendous amount of data. These data are likely to reveal new fundamental facts about life. However, a lot of interesting challenges are left for next-generation biologists, such as to explore the connection between genetic variation and phenotypic variation in humans. In population genetics, questions about populations (of genes) are being addressed by making use of the huge amount of available data.

New computational techniques for collecting and testing hypotheses have been derived for these data, and further methodological development is still needed. The availability of fast computers led to the development of computationally intensive methods, such as the Markov chain Monte Carlo (MCMC). If biologically more realistic assumptions are made, then making inference about unknown parameters in

population genetics often becomes more challenging and time-consuming. The objective of this thesis is to contribute to advanced data analysis problems in modern population genetics. A main problem when analyzing population genetic data is the large number of possible genealogies (see Table 1.1) of the DNA sequence data, This makes inference by full-likelihood methods (see Section 2.2.2) often infeasible.

Table 1.1: Number of possible genealogies depending on the sample sizes (in terms of the number n of sequences)

n	Genealogies
2	1
3	3
4	18
5	180
6	2700
7	56,700
8	1,587,600
9	57,153,600
10	2,571,912,000
100	1.37×10^{284}
1000	3.02×10^{4831}

Therefore Monte Carlo and MCMC methods have been proposed to approximate the full-likelihood for estimating a parameter of interest. In this thesis, we address the problem of estimating the scaled mutation rate has by maximum likelihood and a method is developed that is based on the '*dynamic programming*' approach. This approach computes the exact likelihood to be used for estimating the scaled mutation rate $\theta = 4N\mu$. We consider both the situation where the mutation rate changes over time, i.e., $\theta(t)$ and the case where θ is independent of time.

Approximate Bayesian Computation (ABC) has been introduced to avoid explicit calculation of the likelihood [*Beaumont et al.*, 2002]. Here we propose methods to improve the reliability of ABC. The two main problems we address here, are the choice of the summary statistics, and the choice of the acceptance cutoff. For choosing

summary statistics, a computationally fast and reliable method is investigated that is based on least angle regression. Furthermore, we develop and investigate several algorithms for choosing the acceptance cutoff in the framework of ABC.

1.2 Genetics: Basic Terminology

Although most of the methods presented in this thesis can be applied to non-human genetic data, we will, for simplicity, focus on human genetics. Each human has 46 chromosomes consisting of 23 pairs of chromosomes. Chromosomes are made up of sequences of nucleotides, the deoxyribonucleic acid (DNA) bases. DNA or ribonucleic acid (RNA) molecules carry the whole hereditary information of any living organism. The information is encoded by the four bases adenine, guanine, cytosine, thymine/uracil. These are abbreviated by the letters A, G, C, T/U respectively.

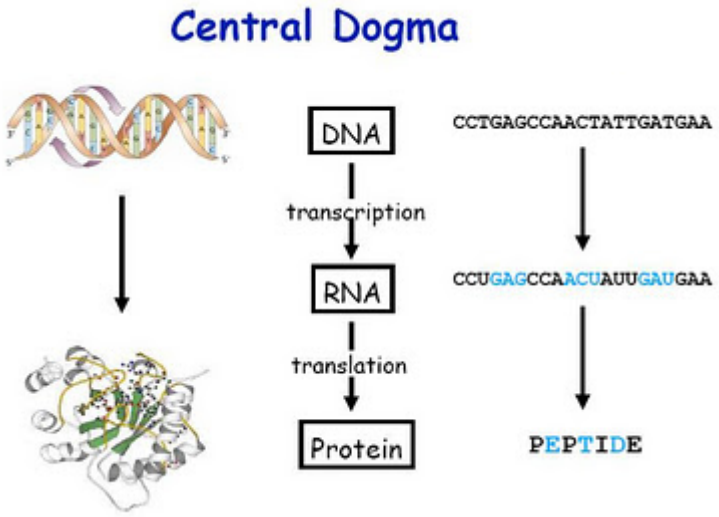

Figure 1.1: Central dogma of genetics

Figure 1.1 is taken from a following URL[1]. Within each cell, the genetic informa-

[1] http://www.betz.lu/index.php/2006/10/27/hitchhikers_guide_to_rna_interference

tion flows from DNA to RNA (transcription) and RNA to Protein (translation). The flow of information is unidirectional and irreversible (see Figure 1.1). It means that the process of producing proteins is irreversible: a protein cannot be used to create DNA.

A gene is a linear region of DNA that controls a hereditary characteristic, and it usually corresponds to a single protein or RNA molecule. Only part of this DNA is functional: the genes. Every gene can be represented by a sequence of A, G, C, T/U, and its size varies between 20 and several 1000 nucleotides. For example, the smallest known gene 'mccA' is only 21 nucleotides base pair (bp) long, and is part of plasmid pMccC7 of Escherichia coli. It encodes unmodified heptapeptide [see *Gonzalez-Pastor et al.*, 1995].

A-T-G-C-G-T-A-C-T-G-G-T-A-A-T-G-C-A-A-A-C

Chromosomes range in size, for instance from 250 million bases (250 Mb) for chromosome 1, to 50 Mb for chromosome 21 in human. The total length of the human genome is approximately 3000 Mb. For each pair of chromosomes, one was inherited from the mother, and the other one from the father. The total hereditary information, also called genome, is transferred from the parental to the filial generation. This involves the process of DNA replication, which leads to predominantly identical copies of the genome. Mutations are changes to the base sequence of genetic material. They are caused, among other reasons, by copying errors induced, for instance, by radioactivity, or ultraviolet radiation. In multicellular organisms, mutations that are transferred to descendants are called germ-line mutations. Germ-line mutations lead to different versions of genes, named alleles. A species consists of all living organisms that are able to interbreed and share some main characteristics. Individuals of a species living in a common geographic area at the same time form a so-called

population. One of the fundamental questions of biology is how populations evolved throughout time. In order to give a description of what evolution is, one has to distinguish between the genotype and the phenotype of an individual. The genotype of an individual is its genome, whereas the phenotype comprises of its biological characteristics. Note that the phenotype is determined by both the genotype and the surrounding environment, as well as by interaction of these two factors. Thus only the genotype is directly inherited. Several biological events (see Section 1.4) change the chromosomes over time when they are transmitted from one generation to another. The events that we consider here, which will have considerable importance in the rest of this thesis, are mutation and recombination.

1.3 Genetic Data and Computer Programs

Consider two types of genetic data: individuals in pedigrees, and in population samples. In both of this cases there is some form of dependency in the data: in a pedigree, all individuals are directly linked; in a population, the sampled individuals are also related, although more distantly, but the nature of their relatedness is unknown. In both cases it is possible to collect phenotypic and genetic information for each individual under study. The phenotypic information usually consists of a trait under study, i.e., the disease status, or another measure of health; genetic information is extracted from biological material, i.e., blood sample. The available genetic information has changed over time, particularly with respect to the amount of data that is available. Today, the information may consist of a short DNA segment, data from thousands of sites across a larger region of a chromosome, or even genome-wide data.

Here we focus on population data that consist of data from a sample of unrelated individuals. The size of such a sample ranges usually from only a few to a few hundred individuals. Whilst unrelated, the genetic data from these individuals will still

be dependent. If we consider a single locus and trace the ancestry of the chromosomes as we go back in time, pairs of individual segments will share a common ancestor at different time-points. The information about these common ancestors can be compactly described through a genealogy. This genealogy determines the dependence in the data at that locus. This dependence, both across loci and across individuals, makes inference from population data challenging. The most common approach to inference is to introduce an appropriate stochastic model for the genealogy (or genealogies) of the sample. When calculating the likelihood, we face a missing-data framework, where the genealogical information is the missing data, and calculation of the likelihood function requires averaging over all possible genealogies for the data.

A genetic marker is a DNA sequence with a known location on the genome; the genetic marker is segregating if there exist variations in the sample at that site. One can imagine a set of genetic markers for one individual to be a sample of the genetic code along a chromosome. The markers that are currently most commonly used exhibit binary variation at a population scale in most cases, and are called single-nucleotide polymorphisms (SNPs). Our focus is on SNP data. For an example of a genealogy of a single locus (chromosome segment) see Figure 1.2.

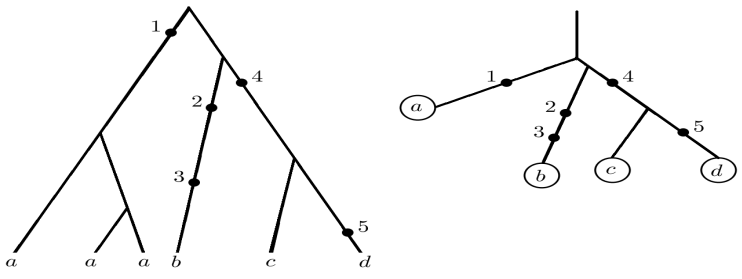

Figure 1.2: Genealogical history (left) and perfect rooted phylogeny (right)

Figure 1.2 taken from *Hobolth et al.* [2008], and genealogical history (left side) of six randomly-sampled haplotypes, with numbered dots representing mutations and letters alleles (distinct nucleotide sequences) and perfect rooted phylogeny (right side)

constructed from the sequence information.

For larger samples the number of possible genealogies is huge, and efficient computer software is therefore needed for the estimation of unknown parameters in population genetics. In our simulations we used programs such as ms [*Hudson*, 2002], and msABC [*Pavlidis et al.*, 2010]. *Excoffier and Heckel* [2006] provide a list of the computer programs for the analysis of specific problems, and they also discuss the limitation of these computer programs.

1.4 Evolutionary Mechanisms

The complete set of unique alleles in a population is called the "gene pool". Large gene pools hint towards an extensive genetic diversity which can sustain a wide range of environmental selection. Similarly, low biological diversity might result in reduced biological fitness and increased chance of extinction. A change in the gene pool of a population over time is called evolution which eventually leads to the formation of new species. Theodosius Dobzhansky in 1973 said, "Nothing in Biology makes sense except in the light of Evolution". Evolution originates from interactions between different processes, which introduce variation into a population and/or remove variation from a population. As a result, variants with a particular trait will become more or less common in a population. The source of variation that leads to evolution, could be genetic as well as environmental.

Mutation (genetic). The main source of variation in population is genetic mutation, which can be passed on to next generation through reproduction. Radiation, viruses, transposons, mutagenic chemicals as well as errors during meiotic processes and *DNA* replication can cause genetic mutations. However, only those changes are passed on to the next generation that occur in egg or sperm cells, or those that occur during or

after fertilization. Mutation might result in a new genotype of descendants.

Recombination (genetic). In eukaryotes, each cell carries two copies of a particular chromosome that are called homologous chromosomes. Each parent passes on one copy to the offspring. During prophase of meiosis I, there is an exchange of genetic material taking place between homologous chromosomes, and the process is called Crossover. Crossover occurs when matching regions on matching chromosomes break and are exchanged. This process of recombination is another source of variation in a population that ultimately leads to evolution.

Horizontal gene transfer (genetic). There are also several mechanisms by which genetic material of an organism is passed into another organism without being the offspring of that organism. Such a process is called Horizontal gene transfer (HGT). A HGT can occur through genetic transformations, bacterial transduction, conjugation and other genetic transfer agents.

Natural selection (environmental). The presence of limited resources lead to competition between organisms for survival and reproduction. The phenomenon of natural selection assumes that individuals with different phenotypes vary in their ability to survive and reproduce in the environment. Consequently, an organism with traits that are advantageous over other organisms tends to pass these traits on more frequently to the next generation. As a result, organisms with such traits become more frequent over time. Natural selection can be classified into abiotic selection and biotic selection. Abiotic selection includes drought, temperature and nutrient content of the soil, etc. On the other hand, biotic selection involves factors that are induced by other individuals, for instance, by competing for food or sexual partners (sexual selection). It is also important to note that initially disadvantageous characteristics may become advantageous once the environmental conditions change (preadaptation).

Genetic drift (environmental). Allele frequencies change from one generation to the next because the alleles in the offspring are passed on randomly from their parents.

There is also randomness involved in determining whether an individual will survive and reproduce. Random changes in allelic frequency from generation to generation are called genetic drift. More precisely, genetic drift involves all non-directed random effects on the gene pool such as through random mating. Genetic drift through random mating will be one of the main stochastic ingredients in the derivation of the so called coalescent process.

External factors (environmental) are for instance climate or natural disasters. Under all these genetic and environmental factors, the gene pool of a population undergoes a change over time that is called evolution. The interplay of mutation and natural selection is very important, particularly when a gene is polyphenic, i.e. when different alleles lead to different phenotypes that may have both positive and negative influence.

1.5 Structure of the Thesis

In this thesis, we focus on some problems of statistical inference in modern genetics, propose new statistical methods, and analyze their performance when analyzing genetic data. In Chapter 2, we first review relevant literature related to mathematical models and statistical inference problems in population genetics. Next, we discuss literature on approximate methods of inference and focus, in particular, on approximate Bayesian computation (ABC). In Chapter 3, we propose a new algorithm to compute the exact likelihood under the infinite sites model and provide simulation results illustrating the performance of this algorithm. Chapter 4 is about choosing summary statistics and the acceptance cutoff for approximate Bayesian computation (ABC). These ingredients are very important for the performance of the method. Finally, we end by summarizing our work and discuss possible future developments in Chapter 5.

This page intentionally left blank

Chapter 2

Literature Review

2.1 Population Genetic Models

In population genetics, we investigate and do analysis about genetic variation in populations. This genetic variation may be in among populations (phylogenetic analysis) or within population; it is affected by several processes such as segregation, mutation, recombination, mating structure, migration, selection and other genetic, ecological, and evolutionary mechanisms (as discussed in Section 1.4). The population (of genes) might be from human, animals, or plants. The genetic variability is preserved under a particulate mode of inheritance as proposed by *Mendel* [1866] and later shown by [*Yule*, 1902; *Hardy*, 1908; *Weinberg*, 1908]. The connection between genetics and natural selection were originated in 1918 through the work of Fisher.

Nowadays, knowledge about molecular biology is quickly increasing by advancement in biotechnology, computing power, and mathematical models. The mathematical models are being increasingly employed to deal with complex mechanisms in biology. Interactions between the mathematical and biological sciences have been increasing rapidly in recent years. There are two types of mathematical models: deterministic and stochastic models. Stochastic models play important role in population genetics to intuitively understand the change in allele frequencies with time.

Subsequently, so many processes are interrelate and govern the evolutionary fate of the population; a proper understanding of the significant processes are necessary in developing good mathematical models and also good experiments. Only important factors are included in mathematical model and exclude the irrelevant one. The main objective of model building is to reveal the answers of those questions that are not anticipated before. Mathematical models have a long history; it was started with elementary mathematics by Gregor Mendel. Afterwards, Francis Galton, Karl Pearson, Ronald A. Fisher, J.B.S. Haldane, Sewall Wright set the criteria for good mathematical modeling and statistical methods [*Bürger*, 2000]

Foundation of theoretical population genetics was built by *Fisher* [1930], *Wright* [1931], and *Haldane* [1932]. Furthermore, many more detailed and sophisticated mathematical models were developed by these and other authors [*Malécot*, 1948; *Kimura*, 1955, 1969; *Kimura and Ohta*, 1973; *Ewens*, 1972] during the period from about 1940 to 1980. These mathematical models were about the evolutionary process and about the maintenance of genetic variation within populations. For detailed literature on mathematical models in population genetics [see *Provine*, 1971; *Bürger*, 2000; *Wakeley*, 2004; *Ewens*, 2004].

2.1.1 Some Basic Models

Fisher and Wright were involved in the elaboration of above population genetics theories. Wright further established that in small populations, evolutionary theory should take account of the sampling effects involved in producing one generation from the previous. He called this effect 'random drift'.

The Wright-Fisher model [*Fisher*, 1930; *Wright*, 1931] assumed that each individuals in the population produce an infinite and equal number of offspring. Thus the population size is finite and constant. Because the population is finite in size and

reproduction is a random process, some individuals may not contribute any offspring while other contribute more to the next generation, which would result in changes in frequency of a particular allele in the population. This random change of genetic lineages forward in time is called genetic drift. Backward in time it is the source of the coalescent process (see Section 2.1.2).

The Moran model [*Moran*, 1958, 1962] is also well studied and widely used in population genetics. It has been important for two reasons. First, in contrast to the Wright-Fisher model, it does not involve a fixed previous generation but has overlapping generations. This means that each individual dies one at a time and is replaced by a new one with probabilities calculated from the system prior to death of individual. Second, it has been important from a mathematical point of view, as many results can be derived exactly under the Moran model, and are available only approximately under the Wright-Fisher model.

The Wright-Fisher and the Moran models have been well explored in population genetics. While the Wright-Fisher model represents perfectly non-overlapping generations, the Moran model represents idealized overlapping generations. Real populations might exist somewhere between these two extremes. When the sample size is large, these models can be well approximated by the coalescent process, see Section 2.1.2 [*Wakeley*, 2009; *Hein et al.*, 2005].

Mutations are the ultimate source of genetic variation; without them there would be no evolution. There are several important models of mutations, we discuss some of them here. In the k-allele model, a gene is assumed to occur in one of a finite number of types, mutations are assumed to occur at a constant rate per individual, and are independent of the current type of the individual. The type created by a mutationic chosen according to a probability distribution. *Kimura and Crow* [1964] introduced the infinite-allele (IA) model, which assumes that every time a mutation occurs, it is to a new allele, never seen before in the population. When the number

of alleles tends to infinity, the IA model can be seen as the limit of the k-allele model. This model is mainly for protein polymorphism. The above models are independent of the type of parents, and they are also known as a parent independent mutation (PIM) models. *Kimura* [1969] also proposed the infinite sites (IS) model that assumes the total number of sites to be large and the mutation rate per site to be very small so that (with good accuracy) whenever a mutant appears it occurs at a previously homo-allelic site. This model is mainly for DNA polymorphisms. *Tajima* [1996] shown that the IA model can be obtained from the IS model and vice versa. Furthermore, many mutation models are convergent to the IA model, when mutation rates are low. These results hold when recombination rates are high in the IS model. The IS model saves computational time because it permits to restrict the number of genealogies compatible with the investigated sequence data. In this thesis, we will assume infinite sites models with or without recombination.

Population growth can be modeled in several ways, such as sudden expansion, exponential growth and logistic growth. The case of population growth or shrinkage will be explained in more detail in this thesis. Take for example exponential growth. Here we and a growth rate g and the population size today N_0, to obtain time dependent population sizes

$$N(t) = N(0)e^{-tg}.$$

Here $N(t)$ is the population size t generation in the past, g is the exponential growth rate, and t measures the time in generations. Hudson, Kingman and others recognized that the standard coalescent can be extended to cover the above growth model by manipulating the time scale. In the standard coalescent the time scale is constant, but in a growing population the time scale is proportional to $N(t)$.

To understand these phenomena it is helpful to know that when the population is small then the rate of coalescence is large, and therefore the time intervals between coalescence events are short. In the case when the population is large, the rate of

coalescence is small, and the time intervals to coalescence are long (see Figure 2.1).

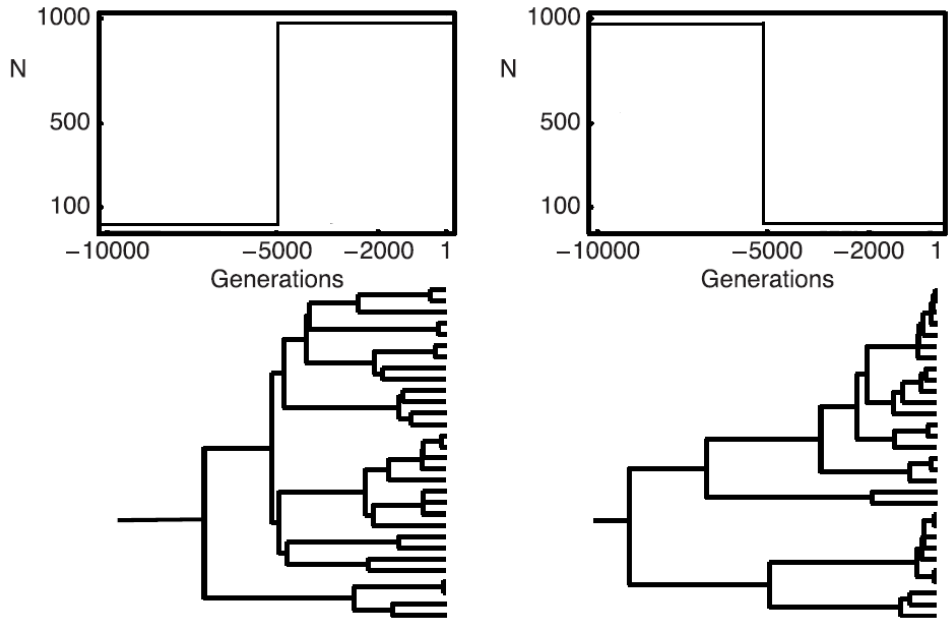

Figure 2.1: Growing and shrinking populations

These phenomena produce on average genealogies for the growing population (left side) with longer branches at the tips and shorter branches at the root than the standard coalescent. For a shrinking population (right side), we observe the opposite picture with shorter branches at the tips and longer branches at the root.

Instead of simply having samples from a single population, one also considers samples from multiple populations. Such a scenario can be modeled by coalescent processes for subdivided populations. Here, we consider coalescence events as well as migration events (events where one lineage moves to the other population). Migration models can have many parameters, for example a simple two population model has at least 4 parameters (see Figure 2.2). Figure 2.1 and 2.2 are taken from a following URL[1].

[1] http://people.sc.fsu.edu/~pbeerli/BSC-5936/11-02-05/lecture_17.pdf

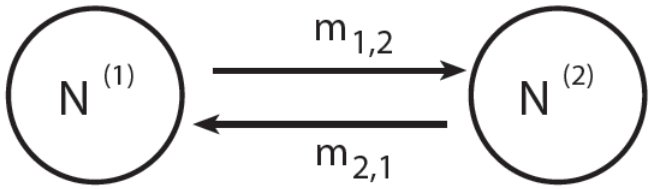

Figure 2.2: Migration model

For two populations we need to consider coalescence in population 1 and 2 and migration events that move lineages from 1 to 2 or 2 to 1.

Gene flow among these discrete populations can be modeled using island models [*Wright*, 1943] and stepping-stone models [*Kimura*, 1953]. There are many further more complicated models in population genetics such as models that involve natural selection (For more information about population genetic models see for instance *Ewens*, 2004.)

2.1.2 The Coalescent Process

As we discussed in previous sections, population genetics relies heavily on mathematical modeling to make quantitative predictions about the behavior of genes in populations. These models are often based on the principles of classic Mendelian gene inheritance, the Hardy-Weinberg equilibrium, and Darwin's theory of natural selection.

We can see in our daily life that the individuals of a population vary in many ways. *Ewens* [1972] proposed a new statistical distribution that predicted patterns of selectively neutral allozyme variation in a sample from a large population. The introduction of the "Ewens sampling formula" marks the beginning of a shift in perspective from a prospective view of classical population genetics to a new, retrospective view

which was soon embodied by the *Kingman* [1982a, b, c] coalescent; [see *Ewens*, 1990; *Wakeley*, 2004] for a discussion of these developments.

If we have the genetic history at our disposal, we could, in principle, understand the variation we observe, but naturally this information is lost, making it impossible to directly predict the present from what we know about the past. Whereas the classical approach uses a forward in time analysis to make predictions about genetic variation in a population and requires a separate theory of sampling, coalescent theory provides a backwards in time approach to generate predictions about genetic variation in a sample. Thus the retrospective approach has always been closely tied to samples and to inference [see *Wakeley*, 2004].

When two copies of a gene descend from the same sequence in a common ancestor, looking back, we say that the copies coalesce (i.e. grow together or join) in that generation. From a retrospective viewpoint, we may then ask which among all possible histories explain our data from a present-day population best. In order to find an answer to this question, one has to model the common history of a population by making assumptions on how evolution works. This is not as restrictive as it might sound at first, because we can include many evolutionary degrees of freedom into the model, which we do not have to specify in advance. These parameters encompass a wide variety of evolutionary aspects such as mutation, selection, recombination, mating structures, or changing population size, and can be estimated from the given data. In this way, statistical methods do shed light on how evolution could have acted on the investigated population.

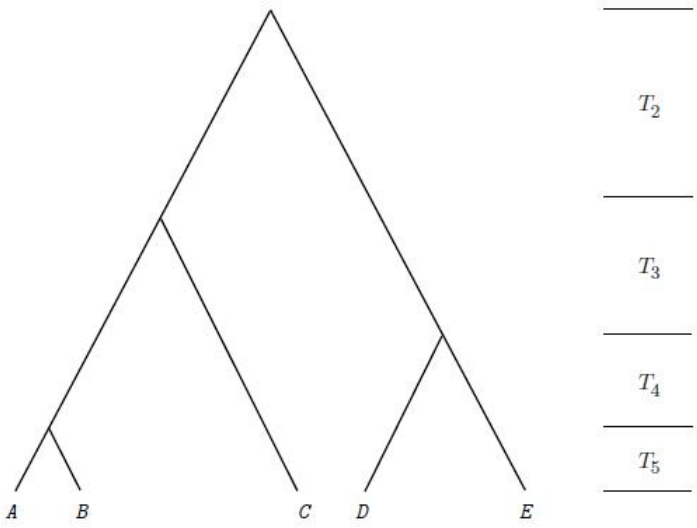

Figure 2.3: One possible genealogy of a sample size five.

Figure 2.3 taken from *Wakeley* [2009]. It shows one possible genealogy of a sample of five gene copies. We assume that the sample is taken at present time, and the genealogical history of the sample is traced back into the past till the most recent common ancestor (*MRCA*). *Kingman* [1982a, b, c] showed that in the limit as the population size $N \to \infty$, the coalescence times T_i are independent and exponentially distributed as

$$f_{T_i}(t_i) = \binom{i}{2} e^{\left(-\binom{i}{2} t_i\right)} \quad t_i \geq 0,\ i = 2, 3, ..., n$$

when time is measured appropriately. The mean and the variance are,

$$E[T_i] = \frac{2}{i(i-1)}$$

$$Var[T_i] = \left(\frac{2}{i(i-1)}\right)^2$$

The length of a lineage is just its vertical height. Typically, T_i is used to designate

the time during which there were i ancestral lineages. On their way up the lineages coalesce, as expected from the facts of *DNA* replication. The coalescent events create junctures, called nodes, which occur at intervals shown to the right. For example, the time back to the first coalescent event in Figure 2.3 is labeled T_5 because during this time there were exactly five ancestral lineages. If n sequences are sampled, then i in T_i ranges from 2 to n. The expected time to MRCA (height of the tree) is given below:

$$E\left[T_{MRCA}\right] = E\left[\sum_{i=2}^{n} T_i\right] = \sum_{i=2}^{n} E\left[T_i\right] = \sum_{i=2}^{n} \frac{2}{i(i-1)} = 2\left(1 - \frac{1}{n}\right)$$

The expected total branch length of the tree is

$$E\left[T_{total}\right] = E\left[\sum_{k=2}^{n} iT_i\right] = \sum_{i=1}^{n-1} \frac{2}{i} \rightarrow 2\left(\gamma + log(n)\right)$$

where $\gamma = 0.57721...$ is Euler's constant. However, in Figure 2.3, the genealogies are drawn without recombination rate, and they are rooted bifurcating trees. Rooted refers to the fact that the deepest branch (uppermost in Figure 2.3) is anchored by the common ancestor of the entire sample. Bifurcating refers to the fact that each node has just three lineages attached to it, one ancestral and two descendant. The root of the genealogy is the MRCA of the entire sample. In the discrete case, the distribution of T_i would be the geometric distribution [*Wakeley*, 2009].

The coalescent process provides good approximations for a wide range of populations with different breeding structures [see *Kingman*, 1982a], and it also provides a model for studying statistical inference in population genetics. Among others, *Hudson* [1983] and *Tajima* [1983] explored many biologically relevant aspects of the coalescent process and presented more intuitive derivations starting with the most basic population model, the Wright-Fisher model. The seeds of the coalescent were planted several decades before this, in the 1940's, by Gustave Malécot, who introduced the idea of following a pair of gene copies back to their common ancestor [*Malècot*, 1946;

Malécot, 1948; *Nagylaki*, 1989; *Slatkin and Veuille*, 2002] and the notion of identity by descent, a concept which is readily interpreted in terms of coalescence events [*Hudson*, 1990]. Genealogical approaches to samples larger than two appeared later, in response to the first direct measurements of molecular variation [*Harris*, 1966; *Lewontin and Hubby*, 1966]. These include *Ewens* [1972] who described the distribution of allele counts in a sample under the infinite-alleles model of selectively neutral mutation, and *Watterson* [1975] who gave an explicitly genealogical derivation of the number of segregating sites, or polymorphic sites, in a sample of sequences under the infinite-sites model of mutation without recombination. In addition, *Griffiths* [1980] theory of lines of descent under the infinite alleles model is based on the coalescent. Lines of descent are sets of descendants of mutations, and *Tavaré* [1984] shows how the structure of the coalescent is recovered from these models by setting the mutation rate to zero. Finally, *Kingman* [2000] draws some connections between the coalescent and earlier work on models of stepwise mutation [*Kimura and Ohta*, 1973; *Moran*, 1975].

The power and popularity of the coalescent derives first from a robustness result, showing that it provides an approximately correct stochastic process for the genealogy of a sample for a wide-range of models of the evolution of a population. Second, the coalescent can be easily extended to various demographic models: for example allowing varying population sizes, and certain forms of non-random mating. It can also be extended to model the joint distribution of genealogies at different loci in the presence of recombination. Lastly, in most cases it is easy and computationally efficient to simulate the coalescent, and hence simulate population genetic data under a range of different modeling assumptions [*Fabrice and Paul*, 2011]. Interested readers should also consult the reviews of coalescent theory by *Hudson* [1990], *Donnelly and Tavaré* [1995], *Möhle* [2000], *Nordborg* [2001], *Hein et al.* [2005], and [*Wakeley*, 2009].

2.2 Statistical Inference in Genetics

In this thesis, we focus on statistical inference based on summary statistics, likelihood-based, and likelihood-free methods.

2.2.1 Summary Statistics

Early approaches used summary statistics to estimate unknown parameters in population genetics. For the mutation rate for example, three well known estimators are available in the literature; the first is based on the number of segregating sites [*Watterson*, 1975], the second on the mean number of pairwise nucleotide differences [*Tajima*, 1989], and the third is based on the number of singletons (*Fu and Li*, 1993a). All of these estimators are unbiased or asymptotically unbiased for θ under the assumption of the IS model, but they may have a big variance. The maximum likelihood estimator (MLE) of θ has been used in the literature to study the efficiency of these estimators [*Fu and Li*, 1993b; *Futschik and Gach*, 2008]. More recently, *Pinheiro et al.* [2010] have studied various estimators of this parameter θ, and investigated their asymptotic behavior as well as comparisons of the distribution's behavior of these estimators through simulations. They have analytically proved that Watterson's estimator [*Watterson*, 1975] and the MLE [*Fu and Li*, 1993b; *Futschik and Gach*, 2008] are asymptotically equivalent with the same rate of convergence to normality.

There are also several methods to estimate the recombination rate that are based on summary statistics and often a method-of-moments approach is used. Furthermore summaries of the site frequency spectrum such as D [*Tajima*, 1989] and H [*Fay and Wu*, 2000], and summaries for linkage disequilibrium (LD), for instance the average pairwise correlation coefficient ZnS [*Kelly*, 1997], are often used. Population differentiation summary statistics such as FST [*Hudson et al.*, 1992b, a; *Slatkin*, 1993] are also popular.

2.2.2 Likelihood Inference

The likelihood function is often used as basis for statistical inference. Quantities derived from the likelihood function provide estimates of unknown parameters, and methods for testing hypotheses and selecting models.

Assume we are considering a parametric model $f(y; \theta)$, which is the probability density function with respect to a suitable measure for a random variable Y. The parameter is assumed to be k-dimensional and the data is assumed to be n-dimensional, often representing a sequence of independent and identically distributed random variables: $Y = (Y_1, ..., Y_n)$. The likelihood function is defined to be a function of θ, proportional to the model density:

$$L(\theta) = L(\theta; y) = cf(y; \theta), \tag{2.1}$$

where c can depend on y but not on θ. Within the context of the given parametric model, the likelihood function measures the relative plausibility of various values of θ, for a given observed data point y.

Coalescent theory (see Section 2.1.2) is strong tool for modeling the distribution of the genealogical tree and do data analysis in population genetics [see *Donnelly and Tavaré*, 1997]. In this thesis, one of the interest is to investigate the mutation mechanism, and the population demography that relate to the genealogy. We address here the problem of performing inference about the scaled mutation rate θ. *Kuhner et al.* [1995] proposed method that is based Markov chain Monte Carlo (MCMC) to estimate the θ. Furthermore, Importance sampling [see for introduction *Ripley*, 1987] method is applied to reduce a large variance of above method [for inference problems in population genetics see *Stephens*, 1999].

We now explain the computation of the coalescent likelihood under the infinite sites model for the classic problem to estimate θ. Under the coalescent model, the

likelihood $Pr\,(y/\theta)$ can be viewed as a summation of probabilities over all possible genealogies, where the data $y = (\mathbf{D},\, \mathbf{z})$ are conditional on θ. Data $\mathbf{D}$ is generated from numerous gene genealogies, where some genealogies are more probable and others are less probable. Likelihood $Pr\,(y/\theta)$ calculation is quite easy, when genealogy parameters (coalescent time and topology) are given in advance [see *Hobolth et al.*, 2008]. Ethier, Griffiths and Tavaré (EGT) recursion is well known for calculating exact likelihood by solving a set of recursions [*Griffiths and Tavaré*, 1994a; *Griffiths and Marjoram*, 1996; *Griffiths and Tavaré*, 1997; *Wu*, 2009]. The EGT equation is given below:

$$\begin{aligned} p\,(\mathbf{D},\, \mathbf{z}) =& \frac{(n-1)}{(\theta+n-1)} \sum_{(k:\mathbf{z}_k \geq 2)} \frac{\mathbf{z}_k\,(\mathbf{z}_k-1)}{n(n-1)} p\,(\mathbf{D},\, \mathbf{z}-\mathbf{e}_k) \\ &+ \frac{\theta}{(\theta+n-1)} \sum_{(k \epsilon A)} \frac{1}{n} p\,(S_k\mathbf{D},\, \mathbf{z}) \\ &+ \frac{\theta}{(\theta+n-1)} \sum_{(k \epsilon B)} \sum_{(j \epsilon C_k)} \frac{1}{n} p\,(R_k\mathbf{D},\, R_k\,(\mathbf{z}+\mathbf{e}_j)) \end{aligned} \tag{2.2}$$

Here θ is the scaled mutation rate, and n is the sample size. Moreover, the k^{th} element is deleted from $\mathbf{D}$ and generate $\mathbf{D}'$ haplotype vector by applying $S_k\mathbf{D}$. The k^{th} item is deleted from $\mathbf{D}$ and generate $\mathbf{D}'$ haplotype vector by applying $R_k\mathbf{D}$. Three main parts of the EGT recursion and its first part resembles the coalescent event, and other two parts are belonged to mutation events. There $\mathbf{e}_k$ to be the vector whose k^{th} bit is 1 and the rest is all 0, and the multiplicity of the n^{th} haplotype in $\mathbf{D}$ is $\mathbf{z}_k$.

Indices of mutable haplotypes are stored in set A, which stay different after deleting their first site and indices of all mutable haplotyes are stored in set B. If, then store these Indices of haplotypes are stored in set C_k where $\mathbf{D}'_k$ is matched with (after mutated at first site) $\mathbf{D}_k$ [*Wu*, 2009].

Wu [2009] have tried to efficiently solved Ethier-Griffiths-Tavaré (EGT) recursion,

which looks genealogical history forward in time. Unfortunately, it is not feasible to solve the EGT recursion for the datasets of useful sample size, and sampling-based techniques are required to calculate the likelihood. The number of genealogies (see Figure 1.1) is the major problem while calculating likelihood $Pr\,(y/\theta)$. Thus, computing $Pr\,(y/\theta)$ can often be challenging under genetic models.

Griffiths and Tavaré [1994b] described a method for approximating the likelihood under the infinite sites model, and they developed a program called *ptree* that can solve the EGT recursion exactly when data is small ($n < 20$) [*Wu*, 2009; *Griffiths and Tavaré*, 1994a]. Afterwards, *Griffiths and Tavaré* [1994b] developed a program called "Genetree". *Felsenstein et al.* [1999] point out that the Griffiths-Tavaré (GT) approach is a version of the importance sampling method. The importance sampling method can handle much larger data in Genetree software. The EGT recursion computation is difficult when the sum of the number of haplotypes and the number of sites exceeds 30 [*Hein et al.*, 2005].

Several computationally intensive methods for the estimation of θ under the infinite sites model rely on Markov Chain Monte Carlo [see *Kuhner et al.*, 1995] or Importance Sampling methods [*Nielsen*, 1998]. In practice, it is important to choose a proposal distribution that promotes an efficient search of the state space [*Hobolth et al.*, 2008]. *Stephens and Donnelly* [2000] provide approximation to the optimal proposal distribution (SD) under the infinite sites model by choosing an allele uniformly at random and perform the unique update implied by the choice of allele. *Hobolth et al.* [2008] also gave proposal distribution, and shown that their proposal distribution is more efficient than GT and SD proposal distributions, in terms of the smaller variance.

Importance sampling method can handle much larger data, and is available as the Genetree software. Based on the Ethier and Griffiths algorithm, a recursion can be constructed to calculate the likelihood of a data set. Unfortunately, it is not feasible

to solve the recursion for data sets of useful size, and sampling-based techniques are required to calculate the likelihood. *Griffiths and Tavaré* [1994b, a] described a method for approximating the likelihood under the IS, and *Felsenstein et al.* [1999] point out that the Griffiths-Tavaré procedure is a version of importance sampling. Proceeding backward in time, a proposal distribution suggests histories of the sample by stepwise reduction of the data set, either by coalescence of two identical genes or by removal of a mutation unique to a single gene. *Stephens and Donnelly* [2000, Theorem 1] characterized the optimal proposal distribution for a large class of models, including the IS, and constructed reasonable approximations to the optimal proposal.

Recently, *Hobolth et al.* [2008] have claimed that neither the GT nor the SD proposal takes into account the number of mutations carried by genetic lineages, one expects that those lineages that have experienced more evolutionary events within a given time period (the time since the most recent common ancestor of the sample) have a higher likelihood of having experienced the most recent evolutionary event. Their proposal distribution is more efficient than an earlier methods [*Stephens and Donnelly*, 2000; *Griffiths and Tavaré*, 1994a] in terms of variance.

2.2.2.1 Extension of Likelihood Inference

There are several alternative approaches that have been developed to deal with more complex problems in population genetics, where full likelihood inference is no longer computationally feasible. These approaches include pseudo and composite likelihoods, conditional likelihood, marginal likelihood, profile likelihood, quasi-likelihood, semi-parametric and non-parametric likelihoods, and empirical likelihood.

It is often possible to write some parameters as functions of other parameters, thereby reducing the number of independent parameters. The function is the parameter value which maximizes the likelihood given the value of the other parameters.

This procedure is called concentration of the parameters and results in the concentrated likelihood function, also occasionally known as the maximized likelihood function, but most often called the profile likelihood function. Unlike conditional and marginal likelihoods, profile likelihood methods can always be used, even when the profile likelihood cannot be written down explicitly. However, the profile likelihood is not a true likelihood, as it is not based directly on a probability distribution, and this leads to some less satisfactory properties. Attempts have been made to improve this, resulting in the modified profile likelihood [*Reid*, 2010]. One interpretation of partial likelihood is that the probability distribution of only part of the observed data is modeled, as this makes the problem tractable and with luck provides an adequate first order approximation. A similar construction was suggested for complex spatial models by *Besag* [1974], using the conditional distribution of the nearest neighbors of any given point, and using the product of these conditional distributions as a pseudo-likelihood function. This was one of a class of such likelihoods now often referred as composite likelihoods, after *Lindsay* [1988].

Composite likelihoods are based upon calculating likelihoods for a subset of the data, and then combining these likelihoods as if each subset of the data were independent. Parameter estimates are constructed by maximizing the resulting composite likelihoods. The first composite likelihood method for estimating recombination rates was introduced by *Hudson* [2001], see also *McVean et al.* [2002]. This is based on combining the likelihood for all pairs of SNPs in the data. In general there are two main motivations for using composite likelihood approaches. The first is computational, as calculating likelihoods for subsets of data is often substantially easier than calculating the full-likelihood for the complete data set. The second is that it avoids the need to model higher order dependencies in the data, and thus gives inferences that are based only on modeling of appropriate marginal or low-dimensional aspects of the data. As data sets get bigger we would expect the importance and use of

composite likelihood methods to also increase [*Varin et al.*, 2011].

Composite likelihoods are extensively used within genetics such as, estimation of recombination rates, association and fine mapping methods, detecting genes under selection, and for inference of demography. Currently, different composite likelihood methods are justified by empirical results (mainly through simulation studies) rather than theoretically. Furthermore, some. questions are still unanswered. For instance, asymptotic distributions of composite likelihood estimators are often unknown. Other unresolved issues are under which conditions the asymptotic distribution will be Gaussian, and how can we consistently estimate the variance of this distribution [*Fabrice and Paul*, 2011]. A quasilikelihood is a function that is compatible with the specified mean and variance relationship. Although it may not exist, when it does it has in fairly wide generality the same asymptotic distribution theory as likelihood functions [*McCullagh and Nelder*, 1989; *Li and McCullagh*, 1994].

2.2.3 Likelihood-free Inference

In recent years, however, a number of different strategies have been developed to address the problem of making complex mathematical models usable for likelihood-based inference. Those include methods that explicitly approximate $p(\mathcal{D}|\theta)$ such as Approximate Bayesian Computing (ABC) [*Beaumont*, 2010; *Csilléry et al.*, 2010], simulated (synthetic) pseudo-likelihoods [*Hyrien et al.*, 2005; *Wood*, 2010] or indirect inference [*Gourieroux et al.*, 1993], and also other methods that allow parametrization without explicitly approximating $p(\mathcal{D}|\theta)$, for example, informal likelihoods [*Beven*, 2006] and Pattern-Oriented Modeling [POM; *Wiegand et al.*, 2003, 2004; *Grimm et al.*, 2005]. Despite different origins and little apparent overlap, most of these methods use the same three essential steps:

1. The dimensionality of the data is reduced by calculating summary statistics of observed and simulated data.

2. Based on these summary statistics, $p(\mathcal{D}|\theta)$, the likelihood of obtaining the observed data $\mathcal{D}$ from the model $\mathcal{M}$ with parameters θ, is approximated.

3. For the computationally intensive task of estimating the shape of the approximated likelihood as a function of the model parameters, state-of-the-art sampling and optimization techniques are applied.

In this thesis we focus on approximate Bayesian computation (ABC).

2.2.3.1 Approximate Bayesian Computation

Unlike classical statistics, Bayesian statistics combines the information from the data with prior information to compute the posterior distribution, the target of Bayesian inference.

$$p(\theta/\mathcal{D}) = \frac{p(\theta)\, p(\mathcal{D}/\theta)}{p(\mathcal{D})} \tag{2.3}$$

From (2.3), where $p(\theta)$ is a prior distribution and $p(\mathcal{D}/\theta)$ is the likelihood with data $\mathcal{D}$. The computation of $p(\mathcal{D})$ involves integration over the parameters and can be difficult to compute. So there are several simulation based methods to compute the posterior. The posterior probability, $p(\theta/\mathcal{D})$ can be written as being proportional to $p(\theta/\mathcal{D}) \propto p(\theta)\, p(\mathcal{D}/\theta)$. Bayesian statistics has a large number of applications from daily life problems to complex problems in population genetics. Methods such as MCMC are are available available for obtaining the posterior in more complex problems. In population genetics the situation is particularly difficult, since also the computation of the likelihood is prohibitively expensive or even impossible for many

realistic models. Approximate Bayesian Computation (ABC) has been introduced to avoid explicit calculation of the likelihood. *Marjoram and Tavaré* [2006] and *Csilléry et al.* [2010] review ABC methods nicely. The basic concept behind ABC has first been introduced by *Tavaré et al.* [1997] for a population genetic problem. A detailed discussion of the ABC method will be given in the next section.

So far, there are many schemes of the ABC method available in the literature. The most basic one is as follows:

Algorithm 2.1: ABC-REJ

---

1. Generate θ from $\pi(.)$
2. Simulate $\mathcal{D}'$ from model $\mathcal{M}$ with parameter θ.
3. Accept θ if $\mathcal{D}' = \mathcal{D}$, and return 1.

Here $\mathcal{D}'$ denotes the simulated data, and $\mathcal{D}$ the observed data. The success of this approach will depend on whether the underlying stochastic model $\mathcal{M}$ is easy to simulate. Furthermore, the practicality of algorithm ABC-REJ depends crucially on the order of magnitude of $P(\mathcal{D})$, because the probability of accepting an observation is proportional to $P(\mathcal{D})$. In cases where the acceptance rate is too small, one might resort to approximate methods such as:

Algorithm 2.2: ABC-REJ

---

1. Generate θ from $\pi(.)$
2. Simulate $\mathcal{D}'$ from model $\mathcal{M}$ with parameter θ.
3. Calculate the distances $\rho(\mathcal{D}', \mathcal{D})$ between $\mathcal{D}'$ and $\mathcal{D}$.
4. Accept θ if $\rho(\mathcal{D}', \mathcal{D}) \leq \epsilon$, and return 1.

This approach requires selection of a suitable distance metric ρ as well as a choice of ϵ. As $\epsilon \to \infty$, it generates observations from the prior, and as $\epsilon \to 0$, it generates from observations from the density $f(\theta/D)$. The choice of ϵ reflects the interplay

between computability and accuracy. For given ρ and ϵ, accepted observations are independent and identically distributed from $f\left(\theta/\rho\left(\mathcal{D}', \mathcal{D}\right) \leq \epsilon\right)$ [see *Beaumont et al.*, 2002; *Marjoram et al.*, 2003].

When $\mathcal{D}$ is high-dimensional, this approach can be impractical as well, and then the comparison of $\mathcal{D}'$ with $\mathcal{D}$ can be replaced by using lower-dimensional summaries of the data. *Weiss and von Haeseler* [1998] extend the method for multiple summary statistics and multiple parameters. Instead of exact match, *Weiss and von Haeseler* [1998] introduce $|\mathbf{S} - \mathbf{S}'| \leq \epsilon$ step for accepting θ, where $|\mathbf{S} - \mathbf{S}'|$ is a distance and ϵ the tolerance parameter.

Algorithm 2.3: ABC-REJ

1. Generate θ from $\pi\left(.\right)$
2. Simulate $\mathcal{D}'$ from model $\mathcal{M}$ with parameter θ,
   and calculate the summary statistics $\mathbf{S}'$.
3. Calculate the distances $\rho\left(\mathbf{S}', \mathbf{S}\right)$ between $\mathbf{S}'$ and $\mathbf{S}$.
4. Accept θ if $\rho\left(\mathbf{S}', \mathbf{S}\right) \leq \epsilon$, and return 1.

Here $\mathbf{S}$ and $\mathbf{S}'$ are observed and simulated summary statistics respectively. The use of summary statistics adds one more layer of error towards approximation. The choice of summary statistics is an important research issue with the *ABC* method [*Marjoram et al.*, 2003]. Ideally, the summary statistics should be sufficient statistics. Since sufficient summary statistics are usually not available in population genetics, *Joyce and Marjoram* [2008] developed a practical method that selects informative summary statistics in *ABC*. *Fu and Li* [1997] were interested in estimating the time to the most recent common ancestor (*MRCA*), one of the key parameters in population genetics. Furthermore, these methods have been adopted by *Wall et al.* [2000], *Tishkoff et al.* [2001] and *Estoup et al.* [2002].

Improvements are proposed by fitting a local-linear regression of simulated pa-

rameter values on summary statistics. Using such a regression relationship, the ABC algorithms results can generally be enhanced by adjusting the i^{th} accepted parameter value θ_i by reducing a discrepancy between its relevant summary statistic $\mathbf{S}'_i$ and the observed value $\mathbf{S}$. *Beaumont et al.* [2002] proposed a method to fit the homoscedastic regression model

$$\theta_i = \alpha + (\mathbf{S}'_i - \mathbf{S})^T \beta + \phi_i$$

replaced θ_i with

$$\theta'_i = \hat{\alpha} + \hat{\phi}_i = \theta_i + (\mathbf{S}'_i - \mathbf{S})^T \hat{\beta},$$

where $(\alpha,\ \beta)$ are vector of coefficient, and weighted least squares estimator of it is $\left(\hat{\alpha},\ \hat{\beta}\right) = \left(\mathbf{X}^T\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{W}\theta$, where $\mathbf{X}$ is the design matrix with $(1,\ \mathbf{S}'_i - \mathbf{S})$ at i^{th} row. The weight matrix was taken to be

$$W_{ij} = \begin{cases} K\left(||\mathbf{S}'_i - \mathbf{S}||\right), & i = j \\ 0 & otherwise. \end{cases}$$

with K the Epanechnikov kernel

$$K_\delta(t) = \begin{cases} 3\left(1 - (t/\delta)^2\right)/(4\delta), & t \leq \delta \\ 0 & t > \delta \end{cases}$$

Beaumont et al. [2002] gave posterior approximations by applying the weights W_{ii} to the θ'_i.

Similarly, the variance can be adjusted using a local log-linear regression for the squared residuals from the mean adjustment:

$$log\left(\hat{\phi}_i^2\right) = \alpha' + (\mathbf{S}'_i - \mathbf{S})^T \beta' + \phi_i$$

Beaumont et al. [2002] also proposed the constant weighted least squares approach

to estimate (α', β').

$$\theta_i'' = \hat{\alpha} + \hat{\phi}_i \frac{\hat{\sigma}(\mathbf{S})}{\hat{\sigma}(\mathbf{S}_i')} = \hat{\alpha} + \hat{\phi}_i exp\left\{(\mathbf{S} - \mathbf{S}_i')\,\hat{\beta}'/2\right\}.$$

[*Blum and François*, 2009] suggested another method for mean and variance adjustments, using Feed-forward neural networks.

There are many schemes of the *ABC* algorithms available. Acceptance rates for the Algorithm 4 can be very low as candidate parameter vectors θ are generated from the prior $\pi(.)$, which may be diffuse with respect to the posterior [see *Marjoram et al.*, 2003]. Accordingly, *Marjoram et al.* [2003] proposed to embed the likelihood-free simulation method within the well known *MCMC* framework. This algorithm proceeds as follows:

Algorithm 2.4: ABC-MCMC

---

1. If at θ propose a move to θ' according to a transition kernel $q(\theta \to \theta')$
2. Simulate $\mathcal{D}'$ from model $\mathcal{M}$ with parameter θ',
   and calculate the summary statistics $\mathbf{S}'$.
3. If the distances $\rho(\mathbf{S}', \mathbf{S}) \leq \epsilon$ between $\mathbf{S}'$ and $\mathbf{S}$ is less than tolerance ϵ
   then go to step 4,
   and otherwise stay at θ and return to step 1.
4. Calculate $h = h(\theta, \theta') = min\left(1, \frac{\pi(\theta')q(\theta' \to \theta)}{\pi(\theta)q(\theta \to \theta')}\right)$
5. Accept θ' with probability h and otherwise stay at θ, then return to step 1.

Algorithm *ABC-MCMC* generates a sequence of serially and highly correlated samples from $f(\theta|\rho(\mathbf{S}', \mathbf{S}) \leq \epsilon)$. The acceptance rates of the *ABC-MCMC* delivers substantial increases over the *ABC-REJ* provided that the prior and posterior are dissimilar, although at the price of generating dependent samples. However, a problem with ABC-MCMC sampler is when it enters an area of relatively low probability with a poor proposal mechanism, the efficiency of the algorithm is strongly reduced.

It is because of difficulty to move anywhere with a reasonable chance of acceptance causing the sampler to stick in that part of the state space for long periods of time *Marjoram et al.* [2003].

However, there is another sampler, which is known as a sequential Monte Carlo (*SMC*) sampler [see *Del Moral et al.*, 2006]. The SMC sampler is used to draw a sample from the easy-to-sample distribution and then moving towards difficult-to-sample distribution by weighting and adjusting in T steps. ABC-SMC sampler is given below:

Algorithm 2.5: ABC-SMC

---

1. Initialize ϵ_1, ϵ_2, ..., ϵ_T, and specify initial sampling distribution μ_1.

   Set population indicator $t = 1$

2.0. Set particle indicator $i = 1$.

2.1. If $t = 1$, sample $\theta^{**} \sim \mu_1(\theta)$ independently from μ_1.

   If $t > 1$, sample $\theta^*$from the previous population $\{\theta_{t-1}^{(i)}\}$

   with weights $\{W_{t-1}^{(i)}\}$, and perturb the particle to $\theta^{**} \sim K_t(\theta|\theta^*)$

   according to a Markov transition kernel K_t.

   Simulate $\mathcal{D}'$ from model $\mathcal{M}$ with parameter θ^{**},

   and calculate the summary statistics $\mathcal{S}'$.

   If $\rho(\mathbf{S}', \mathbf{S}) \geqslant \epsilon_t$, then go to step 2.1

2.2. Set $\theta_t^{(i)} = \theta^{**}$ and calculate the weight for particle $\theta_t^{(i)}$,

$$w_t^{(i)} = \begin{cases} 1, & if\ t = 1 \\ \frac{\pi(\theta_t^{(i)})}{\sum_{j=1}^{N} w_{t-1}^{(i)} K_t(\theta_{t-1}^{(j)}, \theta_t^{(i)})} & if\ t > 1 \end{cases}$$

If $i < N$ set $i = i + 1$, go to step 2.1.

3. Normalize the weights. If $t < T$, set $t = t + 1$, go to step 2.0.

Particles sampled from the previous distribution are denoted by a single asterisk, which changed into a double asterisk after perturbation. The drawback of the SMC algorithm is selection of parameters. In order to address this problem, *Del Moral*

et al. [2010] provided a adaptive *SMC* algorithm for *ABC* with a computational cost that is linear in the number of samples and calculates the tolerance levels adaptively. However, a crucial difference between *ABC-MCMC* and *ABC-SMC* algorithm is that in the ABC-MCMC the current value is the old value if the update is not accepted, otherwise it is the new value, whereas in this *ABC-SMC* algorithm we keep going until we get an acceptance [*Del Moral et al.*, 2010]. So in the *ABC-MCMC* algorithm, one is guaranteed that if the point is already update from the posterior distribution, the next point will also be from the posterior, whereas in the *ABC-SMC* this is not the case [*Beaumont et al.*, 2009].

Sisson et al. [2007] proposed ABC-PRC algorithm that was based on sequential Monte Carlo (SMC) samplers, and it is useful when likelihood computation is prohibitive. They claimed that their proposed method can overcome these inefficiencies. However, *Beaumont et al.* [2009] point out biased weights in *Sisson et al.* [2007] algorithm, which has a visible impact on the quality of the approximation.

Algorithm 2.6: ABC-PRC

1. Initialize ϵ_1, ϵ_2, ..., ϵ_T, and specify initial sampling distribution μ_1.

   Set population indicator $t = 1$

2.0. Set particle indicator $i = 1$.

2.1. If $t = 1$, sample $\theta^{**} \sim \mu_1(\theta)$ independently from μ_1.

   If $t > 1$, sample θ^* from the previous population $\left\{\theta_{t-1}^{(i)}\right\}$

   with weights $\left\{W_{t-1}^{(i)}\right\}$, and perturb the particle to $\theta^{**} \sim K_t\left(\theta|\theta^*\right)$

   according to a Markov transition kernel K_t.

   Simulate $\mathcal{D}'$ from model $\mathcal{M}$ with parameter θ^{**},

   and calculate the summary statistics $\mathbf{S}'$.

   If $\rho\left(\mathbf{S}', \mathbf{S}\right) \geqslant \epsilon_t$, then go to step 2.1

2.2. Set $\theta_t^{(i)} = \theta^{**}$ and calculate the weight for particle $\theta_t^{(i)}$,

$$w_t^{(i)} = \begin{cases} \dfrac{\pi\left(\theta_t^{(i)}\right)}{\mu_1\left(\theta_t^{(i)}\right)}, & if\ t = 1 \\ \dfrac{\pi\left(\theta_t^{(i)}\right) L_{t-1}\left(\theta^*|\theta_t^{(i)}\right)}{\pi(\theta^*) K\left(\theta_{t-1}^{(j)}, \theta_t^{(i)}\right)} & if\ t > 1 \end{cases}$$

where $\pi\left(\theta\right)$ denotes the prior distribution for θ,

and L_{t-1} is a backward transition kernel.

If $i < N$ set $i = i + 1$, go to step 2.1.

3. Normalize the weights so that $\sum_{i=1}^{N} W_t^{(i)} = 1$.

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

   the particles $\left\{\theta_t^{(i)}\right\}$ with weights $\left\{W_t^{(i)}\right\}$ to obtain a new population

$\left\{\theta_t^{(i)}\right\}$, and set weights $\left\{W_t^{(i)} = 1/N\right\}$.

4. If $t < T$, set $t = t + 1$, go to step 2.0.

ABC-SMC algorithm, and the *ABC-PRC* algorithm of *Sisson et al.* [2007] are very similar, in principle, and the main difference is that while *ABC-SMC* is based on a *SIS* framework, whereas Sisson et al. used a *SMC* sampler as a basis for *ABC-PRC*, where the weight calculation is done through the use of a backward kernel. Both

algorithms are explained in detail by *Del Moral et al.* [2006, 2010]. The drawback of the *SMC* sampler is that it does not provide the possibility to use an optimal backward kernel, and it is hard to choose a good one. *Sisson et al.* [2007] choose a backward kernel that is equal to a forward kernel, and it could be a poor choice suggested by *Toni et al.* [2009].

An alternative version *ABC-PMC* is based on genuine importance sampling arguments bypasses this difficulty, in connection with the population Monte Carlo (*PMC*) method of *Cappé et al.* [2004], and it includes an automatic scaling of the forward kernel [see *Beaumont et al.*, 2009]. Moreover, *Del Moral et al.* [2006] gave the theoretical foundation of sequential Monte Carlo method.

Algorithm 2.7: ABC-PMC

1. Given a decreasing sequence of approximation levels $\epsilon_1, \ldots, \epsilon_T$.
2. At iteration $t = 1$,

   For $i = 1, \ldots, N$, repeat

   Simulate $\theta_i^{(1)} \sim \pi(\theta)$, and

   Simulate $\mathcal{D}'$ from model $\mathcal{M}$ with parameter $\theta_i^{(1)}$,

   and calculate the summary statistics $\mathbf{S}'$ until $\rho(\mathbf{S}', \mathbf{S}) < \epsilon_1$

   Set weight $w_i^{(1)} = 1/N$

   Take σ_2^2 as twice the empirical variance of the $\theta_i^{(1)}$'s
3. At iteration $2 \leq t \leq T$,

   For $i = 1, \ldots, N$, repeat

   Pick θ_i^* from the θ_j^{t-1}'s with probabilities $w_j^{(t-1)}$

   generate $\theta_i^{(t)} | \theta_i^* \sim N(\theta_i^*, \sigma_t^2)$ and

   Simulate $\mathcal{D}'$ from model $\mathcal{M}$ with parameter θ_i^*,

   and calculate the summary statistics $\mathbf{S}'$ until $\rho(\mathbf{S}', \mathbf{S}) < \epsilon_t$

   Set $w_i^{(t)} \propto \pi\left(\theta_i^{(t)}\right) / \sum_{j=1}^{N} w_j^{(t-1)} \phi\left(\sigma_t^{-1}\left\{\theta_i^{(t)} - \theta_j^{(t-1)}\right\}\right)$

   Take σ_{t+1}^2 as twice the weighted empirical variance of the $\theta_i^{(t)}$'s

Beaumont et al. [2009] showed the applicability of *ABC-PMC* and compared its performance with *ABC-PRC*. Furthermore, the *ABC-PMC* algorithm is simpler than the *ABC-PRC* algorithm in the sense that it does not require any backward transition kernel and proposes an automatic scaling of the forward kernel. *Wilkinson* [2008] introduces a model error term and emphasis an importance while making statement about reality from model. Moreover, he suggested that it can be possible to generalize approximate sequential Monte Carlo methods in a similar way to that done for the approximate rejection and approximate Markov chain Monte Carlo algorithms.

Blum [2010] presents non-parametric approach to reduce the bias by introducing an estimator of $p(\theta | \rho(\mathbf{S}', \mathbf{S}) \leq \epsilon)$ based on quadratic adjustment unlike linear adjust-

ment proposed by *Beaumont et al.* [2002]. He also highlights the problem of choosing sufficient summary statistics.

Blum and François [2009] proposed a machine learning approach to the estimation of the posterior density, when the number of summary statistics is large. The new approach fits a non linear conditional heteroscedastic regression of the parameter on the summary statistics, and then adaptively improves estimation using importance sampling.

Cornuet et al. [2009] investigated the Adaptive Multiple Importance Sampling (*AMIS*) algorithm and claimed that the improvement brought by this technique is substantial. *Leuenberger and Wegmann* [2010] propose a reformulation of the regression adjustment in terms of a General Linear Model (*GLM*) to estimate the likelihood function and *ABC-GLM* always consistent with the prior distribution. This allows the integration into the sound theoretical framework of Bayesian statistics and the use of its methods, including model selection via Bayes factors.

Lane et al. [2009] also introduced the *ABC-SMC* algorithm that does not need to know advance the number of model parameters. *Sousa et al.* [2009] showed that *ABC* methods can provide reasonably good estimates in a reasonable computational time for the problems in which the choice of summary statistics is not obvious.

More recently, *Fearnhead and Prangle* [2010] showed how to construct appropriate summary statistics for *ABC* in a semi-automatic manner. They proposed that using an extra stage of simulation to estimate the posterior means vary as a function of the data; and then use these estimates of summary statistics within *ABC*. They showed that simulation-based approach to choosing summary statistics could be orders of magnitude more accurate than this alternative, based on empirical results of two examples from the literature.

All above algorithms depend on a good choice of summary statistics about θ [*Nunes and Balding*, 2010] from the dataset. *Marjoram et al.* [2003] also highlighted

the problem of choosing sufficient summary statistics for *ABC*. [Le *Cam*, 1964] gave a definition of approximate sufficient; *Joyce and Marjoram* [2008] developed a practical method that selects informative summary statistics in the context of *ABC*. Their method known as approximate sufficiency (*AS*), and it is based on odds ratio. Nevertheless, it has some limitations: choice of threshold and inclusion order of summary is important [*Nunes and Balding*, 2010].

Partial least squares (*PLS*) regression can be used to choose summary statistics for ABC *Wegmann et al.* [2009]. Leave-one-out cross-validation criterion is also proposed to choose an optimal number of components *Wegmann et al.* [2009]. The R implemented is available in *"pls"* package [*Mevik and Wehrens*, 2007]. Furthermore, *Blum and Tran* [2010] gave an alternative approach for dimension reduction that is based on neural networks.

Nunes and Balding [2010] proposed two methods based on entropy: one is Maximum Entropy (ME), and the other is two-stage (2S) algorithm. We will use the same performance measure in our simulation study as used by *Nunes and Balding* [2010]. The root sum of square error (RSSE) is given below:

$$RSSE = \left(\frac{1}{r} \sum_{i=1}^{N} I_i ||\theta_i - \theta||^2 \right)^{1/2} \tag{2.4}$$

Where N is number of simulation and r is number of accepted observations. If the pair $(\theta_i, \mathbf{S}_i)$ is accepted, then $I_i = 1$, otherwise, $I_i = 0$. The mean of RSSE (MRSSE) is given below:

$$MRSSE = \frac{1}{d} \sum_{j=1}^{d} RSSE(j). \tag{2.5}$$

Where d is number data sets.

Likelihood-free methods have been used in several applications. These include population genetics [*Beaumont et al.*, 2002], wireless communications engineering

[*Nevat et al.*, 2010], quantile distributions [*Drovandi and Pettitt*, 2010], infectious disease epidemiology [*Lopes and Beaumont*, 2010], HIV contact tracing [*Blum and François*, 2009], the evolution of drug resistance in tuberculosis [*Luciani et al.*, 2009], protein networks [*Ratmann et al.*, 2007, 2009], archeology [*Wilkinson and Tavaré*, 2009]; ecology [*Jabot and Chave*, 2009], operational risk [*Peters and Sisson*, 2006], species migration [*Hamilton et al.*, 2005], chain-ladder claims reserving [*Peters et al.*, 2008], coalescent models [*Tavaré et al.*, 1997], sigma-stable models [*Peters et al.*, 2009], models for extremes [*Bortot et al.*, 2007], susceptible-infected-removed (SIR) models [*Toni et al.*, 2009], pathogen transmission [*Tanaka et al.*, 2006] and human evolution [*Fagundes et al.*, 2007].

This page intentionally left blank

Chapter 3

Exact Likelihood Computation

3.1 Introduction

We discussed the problem of estimating the time independent and time-dependent scaled mutation rate in chapter 2. In the former case, the problem has been addressed by applying the Importance Sampling method with three different proposal distributions. Here, we propose an alternative algorithm that also works for models where the mutation parameter θ changes over time. Compared to other methods in the literature, it sometimes also works better when θ is constant. Our algorithm could be seen as a dynamic programming approach.

We first discuss how data are generated under the coalescent. We use this to derive a transition matrix that moves step by step from the configuration of the data to the common ancestor and allows the calculation of the likelihood. Observing the block-structure of this transition matrix, which is caused by the Markov property of the states, we formulate a dynamic programming algorithm for an efficient calculation of the likelihood, first for time-independent θ and then time-dependent $\theta(t)$.

3.2 Dynamic Programming Algorithms for Estimating θ

3.2.1 Data Generation

Let us assume an infinite site model and no recombination. For the generation of data for a sample of size n, under an individual mutation rate μ and $\nu := 1/2N$ per unit time, we first sample a timed coalescence history H using the coalescence process. Afterwards, mutations M are placed on H using the Poisson distribution to obtain HM. Obviously, the number of coalescences is limited to $n-1$, while the number of mutations is, in principle, unbounded [*Stephens and Donnelly*, 2000]. This process allows an efficient generation of the data. However, simulating this process for obtaining the likelihood is inefficient. Indeed, many coalescence histories would be discarded, since they are not compatible with particular data. Therefore, we will describe a process equivalent to the Markov process, whose transitions to the next step only depend on the previous one.Since the minimum of a sample $(X_1, X_2, \ldots, X_n)$ from the exponential distribution with rates $(\lambda_1, \lambda_2, \ldots, \lambda_n)$ are again exponentially distributed with rate $\lambda = \sum_i \lambda_i$, we can first sample from the exponential distribution with rate λ, and then choose among the $\mathbf{X}$ using a multinomial generalization of the Bernoulli distribution with the vector of "probabilities" $p = (\lambda_1/\lambda, \lambda_2/\lambda, \ldots, \lambda_n/\lambda)$.

3.2.2 Estimation of Time-Independent θ

We have a dataset $\mathbf{D}$ of n haplotypes and M segregating sites with unknown μ and ν or simpler with unknown θ (or $\theta = 2\mu/\nu$). Going backwards in time, from $t = i$ to $t = i + 1$ there are in principle two types of events possible in the genealogy:

1. Coalescence. Two identical sequences coalesce into a single sequence with probability proportional to $n(n-1)/2$.

2. Mutation. Change from the derived state to the ancestral state at a site with probability proportional to $n\theta/2$.

We will reach the most recent common ancestor ($MRCA$) after $S = M + n$ such backward steps. The Markov structure of these transitions can be used for an efficient dynamic programming algorithm that computes the likelihood both for time-independent and time-dependent θ.

Algorithm 3.1: Estimating Time-Independent θ

- Initialization ($s = 0$) : Set the configuration at $s = 0$ to the data set and its probability to unity, i.e., $f_1^{s=0} = 1$

- Recursion ($s = 1$, ..., $S - 1$) : Generate all configurations compatible with the data that are one step below the current configurations (i.e., have one fewer haplotype or one fewer mutation, but not both), index them by l, with $1 \leq l \leq L(s)$, if the current step is s. For the step s+1, index the configurations by g, with $1 \leq g \leq L(s+1)$. Compute: $f_l^{(s+1)} = \sum_g f_g^{(s)} p_{gl}^{(s,s+1)}$, where $p_{gl}^{(s,s+1)}$ are the appropriate transition probabilities in the matrix $\mathbf{T}$ from state s to $(s+1)$.

- Termination: The likelihood is $Pr\,(y/\theta) = f_1^{(S)}$.

Thus, for the initial step $s = 1$ and starting out with n haplotypes, we first sample from the exponential distribution with rate $n(n-1)/2 + n\theta/2$, and then choose a coalescence with probability proportional to $n(n-1)/2$ or a mutation within one of the n haplotypes with probability proportional to θ each. For the s^{th} step and conditional on the sample size being $n - 1$, we first sample from the exponential distribution

with rate $(n-i)(n-i-1)/2 + n\theta/2$, and then choose coalescence with probability proportional to $(n-i)(n-i-1)/2$ or a mutation within one of the haplotypes with probability proportional to θ each. This process is repeated until only one haplotype is left. The probabilities at the next step only depend on previous one. Furthermore, as long as we are not assuming θ to vary with time, we can concentrate exclusively on the sequence of steps, ignoring the times (sample from exponential distribution).

If flat priors are chosen for θ, $Pr(\theta) \propto 1$, the posterior distribution would be proportional to the likelihood of the data given θ, i.e., $Pr\,(y/\theta)$. Therefore the approach can also be used to obtain the posterior of θ given the data in such a case, i.e., $Pr\,(\theta/y)$.

3.2.2.1 Toy Example

Consider the following data set with $n = 5$ and only one segregating site, i.e., $M = 1$: two identical haplotypes show the ancestral state and three identical haplotypes show one mutation. Let s index the steps, where $s = 0$ refers to the original configuration of the data and $s = M + n - 1$ to one haplotype in the ancestral configuration or root. Then we have the following sequence of possible configurations c_l^s of the coalescence history of two haplotypes (frequencies of the haplotypes in parentheses) to the root.

Step	ID	haplotype frequencies
0	c_1^0	(2,3)
1	c_1^1	(1,3)
1	c_2^1	(2,2)
2	c_1^2	(1,2)
2	c_2^2	(2,1)
3	c_1^3	(1,1)
3	c_2^3	(3,0)
4	c_1^4	(2,0)
5	c_1^5	(1,0)

Assume discrete times t. At time $t = 0$, we are in the original configuration, i.e., in state c_1^0. From $t = 0$ to $t = 1$, we can move to c_1^1 or c_2^1 or or stay in the same state. The first move occurs with probability $(2 \cdot 1/2) \cdot \nu = \nu$, the second with probability $(3 \cdot 2/2) \cdot \nu = 3\nu$. The probability of staying in the same state is $1 - (5 \cdot 4/2 \cdot \nu + 5\mu)$. The rest of the probability mass, i.e., $6\nu+5\mu$ belongs to moves to a state incompatible with the data. Generalizing to any transition $t = i$ to $t = i+1$, we obtain the following adapted transition matrix:

$$\begin{pmatrix}
1-10\nu-5\mu & \nu & 3\nu & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1-6\nu-4\mu & 0 & \nu & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1-6\nu-4\mu & \nu & \nu & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1-3\nu-3\mu & 0 & \nu & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1-3\nu-3\mu & \nu & \mu & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1-\nu-2\mu & 0 & \mu & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1-3\nu-3\mu & 0 & \nu \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1-\nu-2\mu & \nu \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}$$

There are total nine possible configurations that are compatible with toy example data (see above table) . So we have 9×9 transition matrix. First row corresponds to movement from configuration c_1^0 to c_1^1, c_2^1, or itself and similarly for second row c_1^1 to c_1^2, c_2^2 or itself.

Note that the rows of this transition matrix do not sum to one; the difference

corresponds to transitions incompatible with the data to which we assign probability zero.

Iterating this matrix for $t \to \infty$, will lead to absorption in the state c_1^5. The likelihood corresponds to the sum from $t = 0$ to $t = \infty$ of the probabilities for c_1^5. For these simple data, iteration of the matrix is possible. For more complicated data, the special structure of the matrix can be used for a simplified algorithm. Before we present such an algorithm, we will treat the case when neither ν nor μ depend on time. We can then set $\theta = 2\mu/\nu$ and simplify the transitions probabilities compatible with the data to:

$$\begin{pmatrix}
0 & 1/(10+5\theta/2) & 3/(10+5\theta/2) & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1/(6+4\theta/2) & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1/(6+4\theta/2) & 1/(6+4\theta/2) & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1/(3+3\theta/2) & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1/(3+3\theta/2) & (\theta/2)/(3+3\theta/2) & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & (\theta/2)/(1+2\theta/2) & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/(3+3\theta/2) & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/(1+2\theta/2) \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}$$

We note the block structure of this matrix, where the blocks correspond to the transitions from one step to the next. This leads to the following algorithm for time-independent θ.

3.2.3 Estimation of Time-dependent $\theta(t)$

Next we consider the case of time-dependent μ or ν (or equivalently θ). This may happen if the population is growing or shrinking (see Figure 2.1). We have two events with the probability of mutation $n\mu$ and coalescence $n(n-1)\nu/2$. For a growing population, when the effective population size is small the probability of coalescence is large, and therefore the time intervals to coalescences are short. In the other case, when the population is large the probability of coalescence is small, and the time

intervals to coalescences are long.

Algorithm 3.2: Estimating Time-Dependent $\theta(t)$

- Initialization ($s = 0$ and $t = 0$): Set the configuration at $s = 0$ and $t = 0$ to the data set and its probability to unity, i.e., $f_1^{s=0}(t = 0) = 1$ and all other variables $f_g^{(s+1)}(t)$ and auxiliary variables $h_1^{s=0}(t)$ to 0.

- Recursion over time $t = (1, ..., T)$ for $s = 0$: Generate all configurations compatible with the data that are one step below the current configurations (as above), index them by l, with $1 \leq l \leq L(s = 1)$. Set $f_1^{(s=0)}(t) = f_1^{(s=0)}(t-1)p_{11}^{(s=1)}(t-1,\, t)$, and $h_g^{(s=1)}(t) = f_1(t-1)p_{1g}^{(s=0,\, s=1)}(t-1,\, t)$, $p_{ll}^{s=0} = Pr(c_l^{s=0}/c_l^{s=0}, \nu, \mu(t))$ and $p_{lg}^{(s=0,\, s=1)} = Pr(c_g^{(s=1)}/c_l^{(s=0)},\, \nu,\, \mu(t))$ are the appropriate time-dependent transition probabilities within state $s = 0$ and between states $s = 0$ and $s = 1$.

- Recursion $s = 1$, ..., $S - 1$ and $t = 1$, ..., T : Generate all configurations compatible with the data that are one step below the current configurations (as above), index them by l, with $1 \leq l \leq L(s)$, for the state s, and g, with $1 \leq g \leq L(s+1)$, for the state $(s+1)$. Calculate the functions $f_l^s(t)$ and $h_l^{s+1}(t)$ using: $f_l^s(t) = h_l^s + f_l^s(t-1)p_{ll}^{s,\, s}(t-1,\, t)$, $h_g^{s+1}(t) = \sum_l f_l^s(t-1)p_{lg}^{s,s+1}(t-1, t)$, where $p_{ll}^{s,s} = Pr(c_l^s/c_l^s,\, \nu,\, \mu(t))$ and $p_{lg}^{(s,\, s+1)} = Pr(c_g^{s+1}/c_l^s,\, \nu,\, \mu(t))$ are the appropriate time-dependent transition probabilities within states and between neighboring states.

- Termination: The likelihood is $Pr\,(y/\theta) = \sum_{t=0}^{T} f_1^{(S)}(t)$.

We note that generally only the quotient $\theta(t) = 2\mu(t)/\nu(t)$, but not $\mu(t)$ or $\nu(t)$ alone matter. To keep the maximum time T to a constant value, we therefore assumed only $\mu(t)$ to be time dependent, while ν is assumed constant at all times.

3.3 Simulation Results

We have simulated scenarios for both time independent and time-dependent θ under an infinite sites model without recombination. For time independent θ, we considered samples of size n=10 and 20; furthermore we considered $\theta \in 2, 4, 6$; and $nL \in 2, 5, 10$ independent loci.

For time-dependent θ, we assumed a growing population with one change in the effective population size by a factor of ten at time $0.1 \times 4N$, and where we choose $N = 10^6$. This leads to a current mutation rate $\theta_C = 2.0$, and an ancestral one of $\theta_A = 0.2$. We used the "*ms*" software [*Hudson*, 2002] to simulate data and wrote a Perl program to convert the "*ms*" output into Fasta format.

For time independent θ, we compared our proposed method with the importance sampling method proposed by *Griffiths and Tavaré* [1994a]. For a single locus an implementation of this method is available in the Genetree software package. The adaptation of the method to multiple independent loci is straightforward, and we wrote a Perl program for this purpose. A C++ implementation of our proposed dynamic programming (DP) approach is available in Appendix II.

Table 3.1: Comparison of Griffiths and Tavaré (GT) and our proposed DP method

nL	n		Mutation (θ)											
			$\theta = 2$				$\theta = 4$				$\theta = 6$			
			GT		DP		GT		DP		GT		DP	
			$\hat{\theta}$	T(m)	$\hat{\theta}$	T(m)	$\hat{\theta}$	T(m)	$\hat{\theta}$	T(m)	$\hat{\theta}$	T(m)	$\hat{\theta}$	T(m)
2	**10**	**Mean**	1.63	*0.16*	1.65	*0.01*	3.33	*0.32*	3.99	*0.14*	4.66	*0.40*	5.24	*0.46*
		SD	0.53	*0.02*	0.49	*0.00*	0.73	*0.01*	0.94	*0.16*	1.22	*0.06*	1.52	*0.36*
	20	**Mean**	1.49	*0.28*	1.78	*0.46*	2.86	*0.52*	3.82	*16.04*	4.33	*0.63*	5.28	*95.27*
		SD	0.67	*0.03*	0.37	*0.42*	1.06	*0.09*	0.91	*17.63*	1.22	*0.06*	1.07	*114.9*
5	**10**	**Mean**	1.85	*0.37*	1.99	*0.03*	2.62	*0.70*	3.73	*0.35*	4.35	*0.95*	6.15	*1.77*
		SD	0.38	*0.02*	0.29	*0.01*	0.85	*0.09*	0.87	*0.22*	0.97	*0.09*	0.71	*1.23*
	20	**Mean**	1.49	*0.77*	2.04	*2.06*	2.20	*1.24*	3.70	*31.54*	4.38	*1.70*	6.03	*355.4*
		SD	0.46	*0.12*	0.39	*1.71*	1.14	*0.06*	0.37	*20.10*	0.83	*0.16*	0.68	*189.5*
10	**10**	**Mean**	1.91	*0.73*	1.98	*0.05*	2.99	*1.48*	3.90	*1.39*	3.68	*1.89*	6.06	*5.05*
		SD	0.34	*0.04*	0.38	*0.02*	0.90	*0.12*	0.58	*1.51*	0.67	*0.16*	0.85	*4.04*
	20	**Mean**	1.59	*1.46*	2.07	*5.29*	2.54	*2.66*	4.17	*74.78*	3.90	*3.52*	6.03	*718.8*
		SD	0.34	*0.10*	0.15	*3.64*	0.48	*0.10*	0.28	*27.81*	1.04	*0.35*	0.66	*384.3*

For time-independent θ, Table 3.1 show that our proposed "dynamic programming" approach gives reliable estimates of time independent θ. Although with some limitations that $n \times \theta < 100$, where n is sample size. Our performance is better in terms of accuracy and time. Notice that our dynamic programming approach gives the exact likelihood. We fix the number of simulation to 10^5 in Genetree. Table 3.1 indicates that and Genetree needs billions of simulation runs to achieve the accuracy of our proposed method because Genetree is based on importance sampling [see *Felsenstein et al.*, 1999; *Stephens and Donnelly*, 2000]. The likelihood becomes flatter and Genetree becomes less accurate n or θ increases.

For time-dependent $\theta(t)$, simulation results are given below:

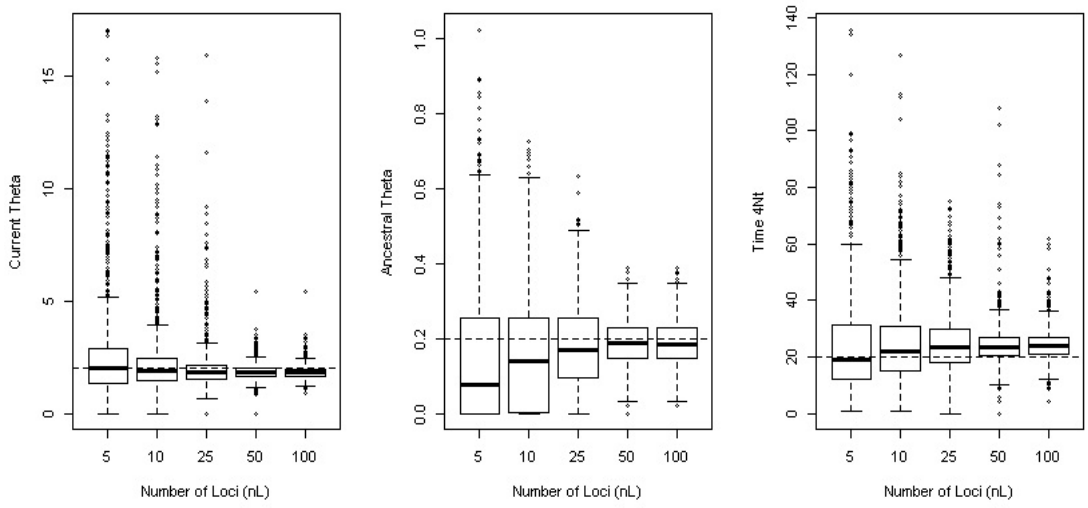

Figure 3.1: Estimates of three parameters for different number of loci (nL= 5, 10, 25, 50, and 100).

Figures 3.1 provide an example of the behavior of the estimates of our three parameters in the model with time-dependent θ. We fixed one parameter and provide contour plots of the marginal likelihood of the remaining parameters. The dotted lines indicate the true parameter values. The contours of the likelihood are skewed due to outliers. Moreover, the mean square error (MSE) for each parameter is given below:

Table 3.2: Mean Square Error (MSE) of growing population

No. of loci	MSE($\hat{\theta}_C$)	MSE($\hat{\theta}_A$)	MSE($\hat{\tau}$)
5	6.337	0.0392	375.41
10	3.149	0.0254	270.87
25	1.241	0.0133	168.75
50	0.387	0.0080	106.74
100	0.126	0.0040	56.85

In Table 3.2, we show the estimated mean squared error (MSE) for each parameter (i.e., current theta ($\hat{\theta}_C$), ancestral theta ($\hat{\theta}_A$), time ($\hat{\tau}$)) based on 1000 simulation

runs. As expected, the MSE decreases with the number of independent loci.

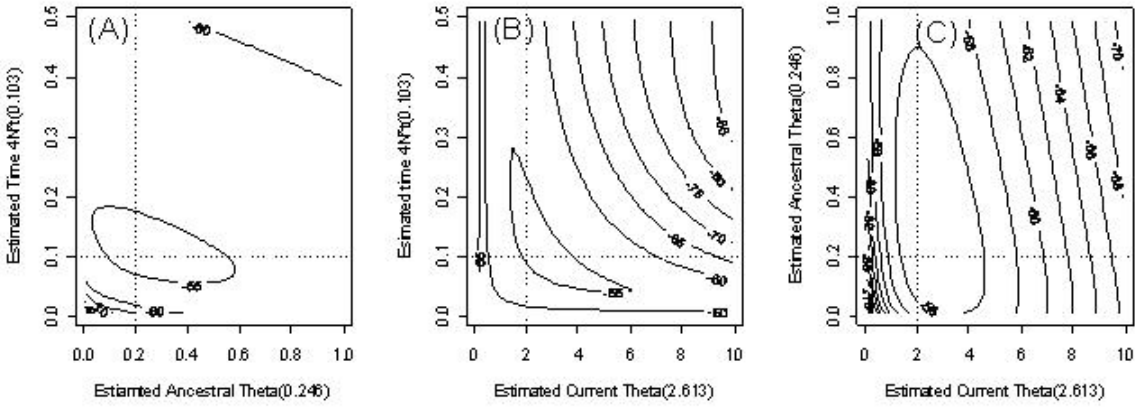

Figure 3.2: Contour plots of three parameters

In the contour plots (see Figure 3.2), the vertical and horizontal dotted lines shows the value of true parameter. We have again three parameters, namely: current theta, ancestral theta and time. We fix one parameter and provide contours plots of the marginal likelihood of the remaining two parameters. We considered five loci in a growing population. In Figure 3.2(A), we fixed the estimated current theta ($\hat{\theta}_C$) to 2.613, the estimated ancestral theta ($\hat{\theta}_A$) is fixed to 0.246 in Figure 3.2(B) and estimated time ($\hat{\tau}$) is fixed to 20.262/200 in Figure 3.2(C). We can conclude from Figure 3.2(A), 3.2(B), and 3.2(C) that our proposed algorithm gives estimates that are close to their true parameter values.

As our proposed dynamic programming algorithm gives the exact likelihood at any parameter value, the application of the algorithm together with an optimization routine will usually provide a local optimum of the likelihood. Here, we used the function Amoeba described in *Press et al.* [2007] to optimize the likelihood.

This page intentionally left blank

Chapter 4

Contributions to Approximate Bayesian Computation

4.1 Introduction

We discussed the literature on approximate Bayesian computation (*ABC*) in chapter 2. In this chapter, the problems of choosing summary statistics and the acceptance cutoff (see step 3 in Figure 4.1) will be addressed. These problems are important in practice, as the efficiency of the algorithms *ABC-REJ*, *ABC-MCMC*, *ABC-SMC*, and *ABC-PMC* all depends on a good solution. In principle, we want to retain as much information as possible from the data. On the other hand, we want to keep the number of summary statistics, and hence the number of dimensions, as low as possible. However, summary statistics in population genetics are usually not sufficient. A statistic is sufficient if it is just as informative as the full data. The concept was introduced by R. A. Fisher in the 1920s, and refined by Jerzy Neyman in the 1930s.

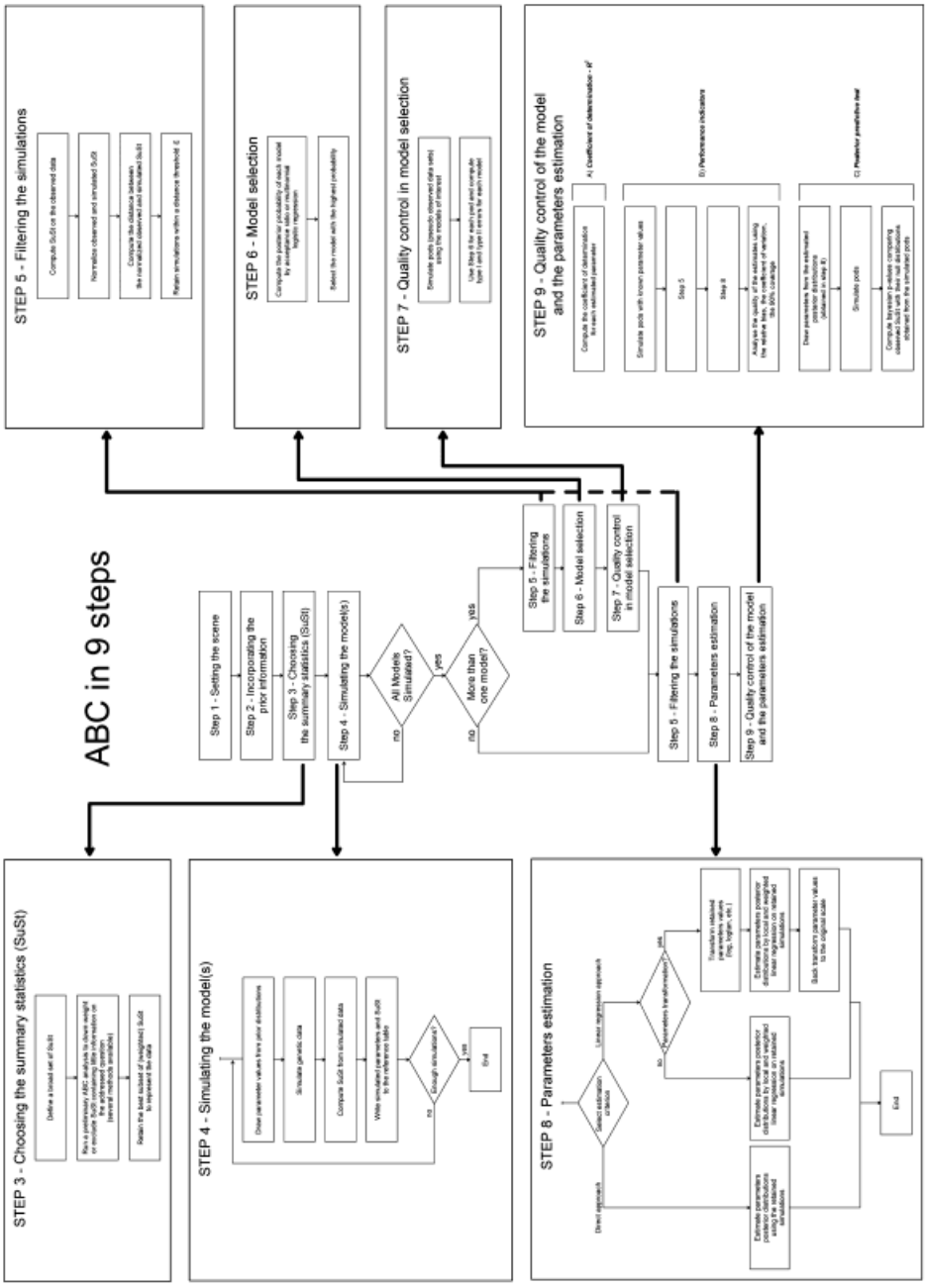

Figure 4.1: Flowchart of ABC in nine steps

Figure 4.1 is taken from *Bertorelle et al.* [2010]. Until now, several methods such as *PLS*, *AS*, *ME*, and two-stage (*2S*) have been proposed that deal with the problem of choosing summary statistics. The limitations of these methods have been discussed in chapter 2. We investigate the use of least angle regression (*LARS*) to

choose the summary statistics for a parameter of interest, and will show that the method performs well in population genetic examples.

Choosing the acceptance cutoff for ABC is also crucial, and there are several methods to deal with this problem. Moreover, we propose and investigate four algorithms for choosing the acceptance cutoff (see Section 4.4).

4.2 Choosing Summary Statistics using LARS

Here we investigated a method for choosing summary statistics that is based on least angle regression (LARS) and cross-validation (CV). First we introduce the methods and afterwards, we investigate the performance of the method when choosing summary statistics. We will investigate the method using the approximate Bayesian computation algorithm given below. We define $\mathbf{S}$ and $\mathbf{S}'$ to be the observed and simulated summary statistics respectively:

Algorithm 4.1: Approximate Bayesian Computation Method

---

1. For $i = 1, ..., N$, repeat

1.1 Simulate parameters $\theta_i \sim \pi(.)$ from the prior distribution;

1.2 Simulate data $\mathcal{D}'$ from model $\mathcal{M}$ with parameter θ_i;

1.3 Calculate the summary statistics $\mathbf{S}' = [S'_1, ..., S'_l]$ from $\mathcal{D}'$;

1.4 Calculate distances $\rho(\mathbf{S}', \mathbf{S})$, where $\mathbf{S} = [S_1, ..., S_l]$;

2. Let $\left[\theta_{[1]}, ..., \theta_{[N]}\right]$ be the sorted values of $\boldsymbol{\theta} = [\theta_1, ..., \theta_N]$;
with respect to their distances $\rho(\mathbf{S}', \mathbf{S})$.

The above algorithm is analogous to Algorithm 2.3, but does not use an acceptance thresholdϵ. Instead we introduce a cutoff r, such that the r closest observations for θ_i with respect to the distances $\rho(\mathbf{S}', \mathbf{S})$. We used the Euclidean distance in all simulation studies that are presented in this chapter.

The least angle regression [*Efron et al.*, 2004] approach is used to choose a list of summary statistics for the parameter of interest. The connection between LASSO and Stagewise become more clear after LARS. With LARS one can not only do variable selection but also can get LASSO solutions easily. *Efron et al.* [2004] proposed the LARS algorithm that is given below:

Algorithm 4.2: Least Angle Regression (LARS)

---

1. Standardize the predictors to have mean zero and unit norm.
Start with the residual vector $\boldsymbol{\phi} = \theta, \quad \hat{\beta}_p = 0 \,\forall\, p$.

2. Find the predictor S_j most correlated with $\boldsymbol{\phi}$.

3. Increase$\hat{\beta}_j$ in the direction of the sign of $corr(\boldsymbol{\phi},\, S_j)$
until some other competitor S_k has as much correlation
with the current residual as does S_j.

4. Update $\boldsymbol{\phi}$, and move $\left(\hat{\beta}_j,\, \hat{\beta}_k\right)$ in the joint least squares direction for
the regression of $\boldsymbol{\phi}$ on $(S_j,\, S_k)$, until some other competitor S_l
has as much correlation with the current residual.

5. Continue in this way until all p predictors have been entered.
Stop when $correlation(\boldsymbol{\phi},\, S_j) = 0 \,\forall\, j$, that is, the OLS solution.

At each step, most correlated predictor is included in model. This process continues until all predictors are in the model [see *Cohen*, 2006]. LARS uses sophisticated angle theory and it is fast. More detail about the least angle regression can be found in [see *Efron et al.*, 2004].

The cross-validation (CV) method is probably the simplest and most widely used method for estimating prediction error. This method directly estimates the average generalization error $Err = E[L(\theta,\, \hat{f}(\mathbf{S}))]$, when the method $\hat{f}(\mathbf{S})$ is applied to an independent test sample from the joint distribution of $\mathbf{S}$ and θ. Ideally, if enough data is available, a validation set would be set a side and can be used to assess

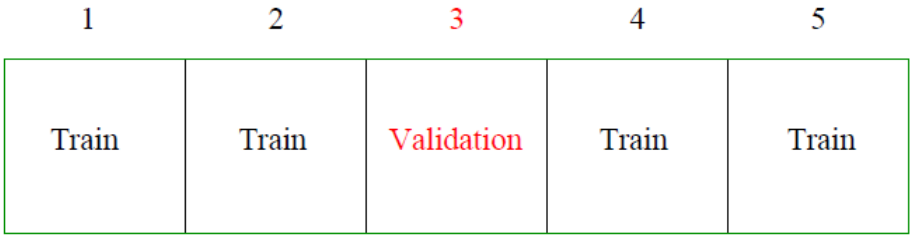

Figure 4.2: Five-fold cross-validation

the performance of prediction model. Since data are often rare, this is usually not possible. To finesse the problem, k-fold cross-validation uses part of the available data to fit the model, and a different part to test it. For this purpose, data is split into K roughly equal-sized parts; for example, when $K = 5$, the scenario looks like this:

For the k^{th} part, the model is fitted to the other $K-1$ parts of the data, and prediction error of the fitted model is calculated while predicting the k^{th} part of the data. This is done for $k = 1, 2, 3, ..., K$ and combines the K estimates of prediction error. Let $\kappa = 1, \ldots, N \rightarrow 1, \ldots, K$ be an indexing function that indicates the partition to which observation i is allocated by the randomization. Denote by $\hat{f}^{-\kappa(i)}(\mathbf{S})$ the fitted function, computed with the k^{th} part of the data removed. Then the cross-validation estimate of prediction error is

$$Err \cong \frac{1}{N}\sum_{i=1}^{N}\left(\theta_i - \hat{f}^{-\kappa(i)}\left(\mathbf{S_i}\right)\right)^2$$

Typical choices of K are 5 or 10 [see *Breiman and Spector*, 1992; *Kohavi*, 1995]. The case $K = N$ is known as leave-one-out cross-validation. In this case $\kappa(i) = i$, and for the i^{th} observation the fit is computed using all the data except the i^{th}. For more detailed information about cross validation procedures see *Hastie et al.* [2009].

The proposed algorithm for choosing summary statistics is given below:

Algorithm 4.3 : Choosing summary statistics for ABC

1. Given a sorted parameter values $\left[\theta_{[1]}, ..., \theta_{[N]}\right]$, and simulated summary statistics $\mathbf{S}' = \left[S'_1, ..., S'_p\right]$ from Algorithm 4.1.

2. Let $\boldsymbol{\theta}^*$ **:=**$\left[\theta_{[1]}, ..., \theta_{[r]}\right]$, where r is user define cutoff.

3. Apply LARS (Algorithm 2) with the a model $f\left(\boldsymbol{\theta}^*|\mathbf{S}'\right) = \alpha + \beta\mathbf{S}' + \phi$, where $\boldsymbol{\theta}^*$ is the response, and $\mathbf{S}'$ are the predictors in the model, and α is intercept and $\beta = [\beta_1, \beta_2, ..., \beta_p]$ are slopes, and the residuals ϕ

4. Define $x_j := \frac{j}{m}$, $1 \leq j \leq m$, and m is user define proportion of the model

5. The CV prediction error is $\hat{R}(x_j) \cong \frac{1}{r}\sum_{i=1}^{r}\left(\theta_{[i]} - \hat{f_{x_j}}^{(-i)}\left(\boldsymbol{\theta}^*|\mathbf{S}'_{[\mathbf{i}]}\right)\right)^2$
Where $\hat{f_{x_j}}^{(-i)}\left(\boldsymbol{\theta}^*|\mathbf{S}'_{[\mathbf{i}]}\right)$ is the predicted values where i^{th} observation is excluded at each part x_j of the model.

6. Define $\hat{R}(x_{min}) := arg\,min_j\left[\hat{R}(x_j)\right]$;
Calculate $x_j^* = arg\,max_{x_j}\hat{R}(x_j) \leq \hat{R}(x_{min}) + \widehat{S.E.}\left[\hat{R}(x_{min})\right]$,

7. At cutoff x_j^*, $\left|\hat{\beta}_p(x_j^*)\right| = \begin{cases} > 0 & select\, S'_p \\ else & reject\, S'_p \end{cases}$

In principle $m = p$ because at each step LARS add one predictor, but we used interpolation method to convert the coefficients β's of p steps into m equal parts. The leave-one-out cross-validation (LOOCV) is used to estimate the prediction error in above algorithm. However, we would recommend the k-fold cross-validation (CV) to estimate the predication error. While the k-fold CV computes the mean squared prediction error for $LARS$ at each of m parts of the model. We denote them by x_j, and its limit is between 0 and 1.

When number of summary statistics is less than 10 then it is straight forward to use the *Algorithm 4.3*. If number of summary statistics is greater than 10 then we iterate the *Algorithm 4.3*. The summary statistics is selected, where the mean square prediction error will be steep. Furthermore, we could also apply C_p criteria to find the important summary statistics for each parameter, but it is not working well with

our objective of choosing summary statistics for ABC algorithm. For the problem of choosing summary statistics, the k-fold cross-validation is performing better than C_p. We implemented the LARS in R, and used the 10-fold cross-validation for estimating the predication errors. Furthermore, stopping criterion at step 6 of the Algorithm 4.3 is '*1 SE Rule*' [see *Breiman et al.*, 1984; *Hastie et al.*, 2009].

We evaluate the performance of our least angle regression *(*LARS*)* based approach, relative to AS, PLS, ME, two-stage (2S) methods, in estimating the unknown population genetics parameters. We will illustrate the approach with two examples in next sections.

4.3 Simulation Results

4.3.1 Example 1

The setup of our simulation study is similar to studies done previously [see *Joyce and Marjoram*, 2008; *Nunes and Balding*, 2010]. The parameters are the scaled mutation and recombination rates, θ and ρ respectively, and the data sets consist of 50 haplotypes being generated by using the *ms* software [*Hudson*, 2002] under the standard coalescent model, following the infinite-sites (IS*)* model [*Nordborg*, 2007]. The prior distribution is the following for the scaled mutation rate $\theta \sim U(2, 10)$ and for recombination $\rho \sim U(0, 10)$. The simulated and prior distribution are chosen same. The seven summary statistics have been calculated, and data were analyzed using the R package named "ABCME" [see *Nunes and Balding*, 2010]. Algorithms such as PLS, AS, ME, two-stage (2S) and LARS have been implemented. Our parameters have been: number of ABC simulation runs $N = 10^6$, number of accepted observations $r = 10^4$, and number of observed datasets $d = 10^2$ for inference of θ and ρ using ABC (see Algorithm 4.1). The results are given below without regression adjustment, and

we use an 1% acceptance cutoff for comparison of MRSSE with other methods. In this thesis, all simulation studies are used the Euclidean distances as distance metric ρ for ABC.

We used the R package "LARS" to implement our proposed method. The results are given below for a smaller example, where the number of iteration is $N = 10^4$, $c = 2000$ and $m = 100$.

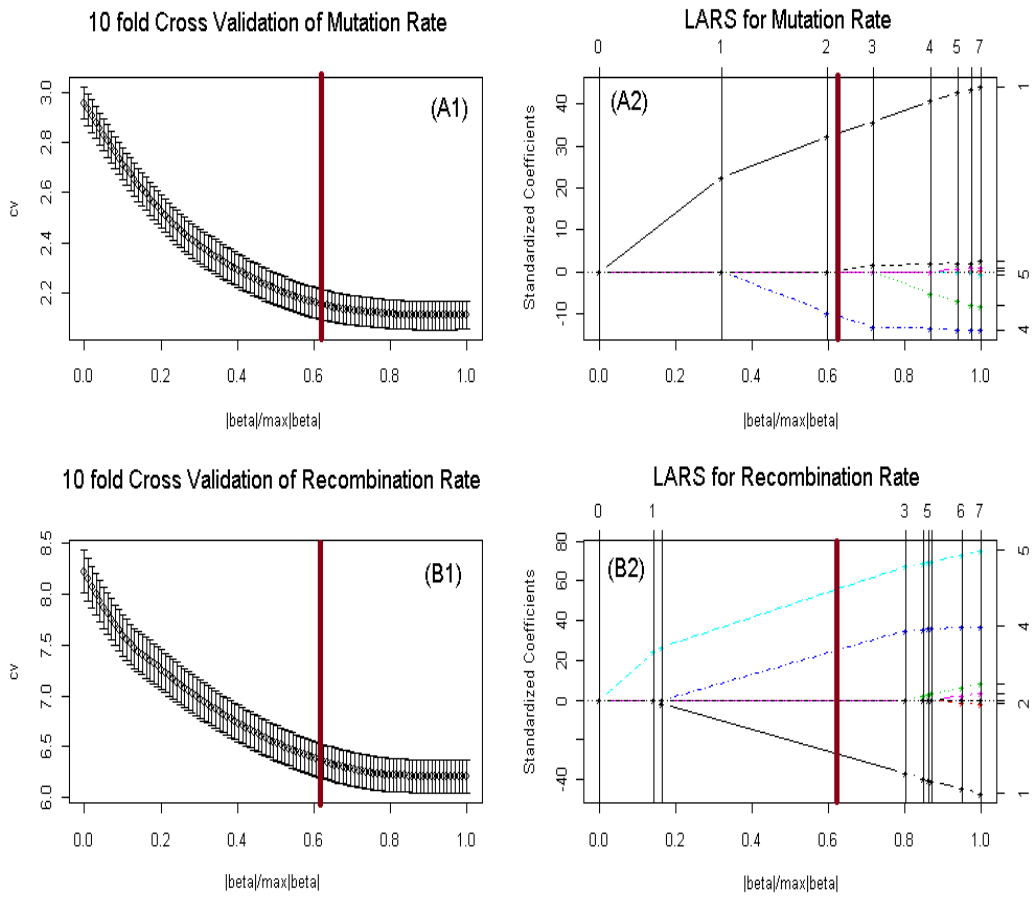

Figure 4.3: Choosing summary statistics for Mutation and Recombination Rate by using LARS (one ABC RUN)

Figure 4.3 shows the output from R package "lars". We see that for the mutation rate C1 and C4 are important (see Figure A1 and A2), and for the recombination rate C1, C4, and C5 are important (see Figure B1 and B2) by using the 10-fold cross validation. It was only one run of ABC. In Figure 4.3, until the vertical dark-red line

the predictors are important for specific parameter,i.e, for mutation important summary statistics are C1 and C4, and for recombination C1, C4, and C5 are important summary statistics.

The set of all summary statistics is shown in Table 4.1. The results for *AS* is taken from [*Joyce and Marjoram*, 2008], and results for *ME*, two-stage (*2S*) are computed in *R* package "*ABCME*" by [*Nunes and Balding*, 2010]. The summary statistics C2 is uniform random variable and it should not be included in an optimum set of summary statistics. The summary statistics C1 (number of segregating sites) for θ [*Hudson*, 1990; *Nordborg*, 2007] and C5 (number of distinct haplotype) for ρ are important summary statistics; they should be included in an optimal set of summary statistics [*Nunes and Balding*, 2010].

Table 4.1: Comparison of difference methods

Statistic	**Description**	Selected for θ (%)				Selected for ρ (%)			
		AS	**ME**	**2S**	**LARS**	**AS**	**ME**	**2S**	**LARS**
C1	No. of segregating sites	75	67	100	100	73	67	97	100
C2	Uniform [0,25] random variable	4	3	0	0	2	5	0	0
C3	Mean no. of differences over all pairs of haplotypes	27	54	25	9	52	30	19	27
C4	25*(mean r2 across pairs separated by <10% of the simulated genomic region)	56	35	50	43	35	59	78	89
C5	No. of distinct haplotypes	43	19	20	17	78	73	100	100
C6	Frequency of the most common haplotype	36	20	1	0	11	23	2	2
C7	No. of singleton haplotypes	16	14	5	1	16	31	5	1

In Table 4.1, the set of all considered summary statistics can be found. There are 100 observed data sets and each summary statistic selected (out of 100) in the optimal set. From Table 4.1, the *AS* method relatively better than *ME*, but the two-stage (*2S*) and *LARS* algorithm performance is better than AS method.

Table 4.2 shows the MRSSE for each method. The results for *AS* is taken from [*Joyce and Marjoram*, 2008], and results for *ME*, two-stage (*2S*) are computed in *R* package "*ABCME*" by *Nunes and Balding* [2010].

Table 4.2: Performance of PLS, AS, ME, 2S, and LARS methods, by MRSSE.

PAR	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**All6**	**PLS**	**AS**	**ME**	**2S**	**LARS**
θ	1.75	3.27	2.26	3.15	2.33	2.89	2.45	1.89	1.85	1.86	1.80	1.70	1.75
ρ	3.93	3.95	3.93	3.92	3.83	3.84	3.88	3.60	3.56	3.68	3.54	3.44	3.46

In Table 4.2, the performance of different methods is shown in terms of the *MRSSE* (see equation 2.5). First seven columns (C1-C7) show the results for single summary statistics; column eight (All6) show results of six summary statistics altogether, expect C2. Last five columns show the results of each method. Three PLS components lead to the smallest *MRSSE.* From Table 4.2 we can conclude that the set of summary statistics selected by *2S* and *LARS* produce better results than other methods.

Table 4.3: Comparison of MRSSE

Cutoff (%)	Adjustment	Mutation (θ)				Recombination (ρ)			
		PLS	ME	2S	LARS	PLS	ME	2S	LARS
0.01	No Adj	1.786	1.784	1.686	1.743	3.525	3.456	3.324	3.342
	Mean	1.763	1.777	1.683	1.738	3.510	3.446	3.301	3.291
	Mean + Var	1.755	1.781	1.682	1.738	3.501	3.419	3.279	3.261
0.05	No Adj	1.824	1.789	1.696	1.751	3.545	3.514	3.399	3.425
	Mean	1.771	1.777	1.686	1.743	3.518	3.485	3.331	3.314
	Mean + Var	1.750	1.777	1.683	1.740	3.487	3.414	3.275	3.240
1	No Adj	1.849	1.796	1.704	1.754	3.559	3.544	3.439	3.464
	Mean	1.776	1.779	1.688	1.743	3.524	3.506	3.344	3.320
	Mean + Var	1.747	1.774	1.681	1.738	3.484	3.407	3.266	3.230
2	No Adj	1.892	1.811	1.720	1.766	3.579	3.582	3.489	3.521
	Mean	1.786	1.789	1.693	1.747	3.530	3.528	3.354	3.327
	Mean + Var	1.745	1.773	1.682	1.737	3.478	3.394	3.251	3.220
3	No Adj	1.925	1.824	1.734	1.776	3.593	3.609	3.527	3.561
	Mean	1.793	1.797	1.696	1.750	3.535	3.542	3.363	3.332
	Mean + Var	1.741	1.770	1.680	1.737	3.475	3.388	3.246	3.215
4	No Adj	1.955	1.838	1.748	1.786	3.605	3.630	3.557	3.591
	Mean	1.799	1.804	1.701	1.753	3.538	3.553	3.371	3.336
	Mean + Var	1.739	1.769	1.679	1.737	3.473	3.387	3.244	3.315
5	No Adj	1.981	1.850	1.761	1.795	3.614	3.648	3.582	3.614
	Mean	1.805	1.811	1.704	1.756	3.540	3.563	3.377	3.338
	Mean + Var	1.737	1.767	1.678	1.736	3.470	3.387	3.243	3.214
10	No Adj	2.089	1.905	1.820	1.839	3.649	3.707	3.665	3.693
	Mean	1.827	1.841	1.719	1.769	3.550	3.594	3.393	3.346
	Mean + Var	1.737	1.758	1.672	1.733	3.466	3.394	3.248	3.222

In Table 4.3, *the MRSSE* (see equation 2.5) for four methods of choosing summary statistics (PLS, AS, ME, and 2S) can be found. We considered different acceptance cutoff percentages. For both parameters, mutation (θ) and recombination (ρ), we applied ABC with and without regression adjustment. For Mean Adjustment [see *Beaumont et al.*, 2002], and for Variance Adjustment [see *Blum and François*, 2009].

4.3.2 Example 2

Our simulation study is about the estimation of the four parameters: mutation rate θ, recombination ρ, migration θ_m, and time at which sub-population 2 will be change into sub-population 1 η_c. The *ms* [*Hudson*, 2002] software is used to generate data sets that consist 50 haplotypes. The prior distribution for the parameters are $\theta \sim U(0,\ 10)$, $\rho \sim U(0,\ 10)$, $\theta_m \sim U(0,\ 0.4)$, and $\eta_c \sim U(0.5,\ 0.9)$. The simulated and prior distribution are chosen same. Twenty-nine summary statistics have been calculated using *msABC* [see *Pavlidis et al.*, 2010], and the three uniform random variables (see Appendix) are added to this set of summary statistics. The algorithms *ME*, two-stage (*2S*), and AS are too time consuming in this example. We therefore compared only PLS with the *LARS* method. Our number of simulation runs has been chosen $N = 10^6$, the number of accepted observations was $r = 500$, and the number of different data sets $d = 10^2$. Thus we tried ABC with $N = 10^6$ runs on each of $d = 10^2$ data sets. As before, we use the Euclidean distance as our metric ρ.

Table 4.4: Optimal set of summary statistics chosen by LARS

Summary statistics		% Selection for each parameter (out of 100)			
		θ	ρ	θ_m	η_c
Prior		U(0, 10)	U(0, 10)	U(0, 0.4)	U(0.5, 0.9)
C1	s_segs_1	0	0	0	0
C2	s_segs_2	0	0	0	0
C3	s_segs	4	0	0	0
C4	s_pi_1	0	0	0	0
C5	s_pi_2	0	0	0	0
C6	s_theta_pi	0	0	0	0
C7	s_theta_w_1	0	0	0	0
C8	s_theta_w_2	0	0	0	9
C9	s_theta_w	100	8	12	38
C10	s_tajimasD_1	8	32	92	10
C11	s_tajimasD_2	4	40	93	11
C12	s_tajimasD	13	77	94	84
C13	s_ZnS_1	1	77	45	1
C14	s_ZnS_2	3	79	51	1
C15	s_ZnS	4	74	56	0
C16	s_fst	2	29	97	0
C17	s_perc_shared_1_2	1	10	6	0
C18	s_perc_private_1_2	5	23	0	0
C19	s_perc_fixed_dif_1_2	0	19	0	0
C20	s_pairwise_fst_1_2	0	10	1	0
C21	s_FayWuH_1	3	41	1	0
C22	s_FayWuH_2	19	37	1	25
C23	s_FayWuH	19	22	1	26
C24	s_dvk_1	1	23	2	0
C25	s_dvh_1	0	1	14	0
C26	s_dvk_2	16	16	8	1
C27	s_dvh_2	0	0	7	0
C28	s_dvk	82	41	4	0
C29	s_dvh	77	0	10	0
C30	Random1	0	0	0	0
C31	Random2	0	0	0	0
C32	Random3	0	0	0	0

In Table 4.4, we show the set of all summary statistics considered in the example 2. There are 100 observed data sets and each summary statistic selected (out of 100) in the optimal set.

According to Table 4.4, our implementation of the *LARS* algorithm chose appro-

priate summary statistics for the mutation parameterθ and the recombination parameter ρ. It never included non-informative (i.e., C30, C31, C32) summary statistics. The summary statistics chosen were those known to be informative for the respective parameters [see *Pavlidis et al.*, 2010]. For the other two parameters migration θ_m, and η_c the performance of LARS has been slightly weaker, probably since these parameters are more difficult to estimate and good summary statistics are not available. For example 2, Table 4.5 contains further results for *PLS* and LARS:

Table 4.5: Comparison of PLS and LARS methods, by MRSSE.

	Summary statistics	θ	ρ	θ_m	η_c
C1	s_segs_1	1.875	3.479	0.148	0.151
C2	s_segs_2	1.893	3.480	0.149	0.152
C3	s_segs	1.528	3.488	0.153	0.152
C4	s_pi_1	2.025	3.484	0.148	0.151
C5	s_pi_2	2.058	3.456	0.149	0.151
C6	s_theta_pi	1.733	3.468	0.153	0.148
C7	s_theta_w_1	1.876	3.479	0.148	0.151
C8	s_theta_w_2	1.894	3.480	0.149	0.152
C9	s_theta_w	1.528	3.488	0.153	0.152
C10	s_tajimasD_1	3.023	3.480	0.152	0.152
C11	s_tajimasD_2	2.961	3.485	0.152	0.153
C12	s_tajimasD	3.113	3.470	0.153	0.149
C13	s_ZnS_1	2.959	3.398	0.151	0.153
C14	s_ZnS_2	2.951	3.418	0.151	0.152
C15	s_ZnS	3.006	3.446	0.152	0.151
C16	s_fst	3.167	3.514	0.148	0.154
C17	s_perc_shared_1_2	2.296	3.443	0.132	0.155
C18	s_perc_private_1_2	2.213	3.563	0.145	0.151
C19	s_perc_fixed_dif_1_2	3.006	3.507	0.145	0.155
C20	s_pairwise_fst_1_2	3.167	3.514	0.148	0.154
C21	s_FayWuH_1	3.077	3.483	0.151	0.153
C22	s_FayWuH_2	3.122	3.525	0.153	0.153
C23	s_FayWuH	3.196	3.515	0.152	0.153
C24	s_dvk_1	2.089	3.229	0.151	0.152
C25	s_dvh_1	2.307	3.296	0.151	0.152
C26	s_dvk_2	2.187	3.301	0.151	0.152
C27	s_dvh_2	2.353	3.354	0.151	0.152
C28	s_dvk	1.899	3.202	0.152	0.152
C29	s_dvh	2.084	3.289	0.152	0.152
C30	Random1	3.168	3.502	0.152	0.152
C31	Random2	3.168	3.502	0.152	0.152
C32	Random3	3.174	3.516	0.152	0.153
All 29	Without C30, C31, C32	1.579	3.060	0.134	0.152
PLS	Five PLS components	1.595	3.119	0.132	0.153
LARS	see Table 4.4	**1.536**	**3.042**	**0.129**	**0.149**

In Table 4.5, we present the *MRSSE* (equation 2.5) for example 2. We consider for each of the 32 summary statistics separately (C1 to C32), as well as all 29 summary statistics other than C30, C31, C32. We then compare the results for such a choice

of summary statistics with the results obtained when choosing summary statistics by PLS and LARS. Bold indicates the lowest value in each column. Partial least squares (PLS) worked best when five components are used. From Table 4.5 we can conclude that the set of summary statistics selected by *LARS* producing better results than PLS method. Indeed, *LARS* gives the lowest *MRSSE* in each column of Table 4.5.

4.4 Choosing an Acceptance Cutoff for ABC.

For all ABC samplers, the choice of tuning parameters is crucial for the quality of posterior estimates. If the tuning parameter $\epsilon = 0$ is chosen, then the algorithm will usually be computationally too expensive and will sample very few point from a posterior distribution. On other hand, if the tuning parameter $\epsilon = \infty$ is chosen, then the samples would be from the prior distribution and not from the posterior. Actually, there is a trade-off between accuracy and computational time. Choosing the tuning parameter can be parsed in terms of selection of an acceptance cutoff. Here we consider two popular methods, leave-one-out cross validation (LOOCV), and one percent cutoff (FIX01). The LOOCV approach is implemented in the R package "*abc*" by Michael Blum. The LOOCV is quite computationally intensive, so we consider it with only 50 cross-validation sample. With FIX01, we select the 1% of the simulated samples for which the distance between observed and simulated summary statistics is smallest. We investigate also four new algorithms for estimating the acceptance cutoff. The algorithms are given below:

Algorithm 4.4 : choosing cutoff for ABC

1. Simulate a vector of parameter N values $\boldsymbol{\theta}^* := \left[\theta_{[1]}, ..., \theta_{[N]}\right]$, sorted by Algorithm 4.1 according to $\rho(\mathbf{S}', \mathbf{S})$.

2. Specify search space for cutoff $r \in \{g+1, g+2, ..., G\}$, where g and G are lower bound and upper bound respectively. They are user defined numbers.

3. Specify validation sample, $\boldsymbol{\theta}^v = \left[\theta_{[1]}, ..., \theta_{[g]}\right]$

4. For $i = 1, ..., g$, repeat

$$RSSE\,(t,\, i) = \left(\frac{1}{t-g}\sum_{j=g+1}^{t}\left(\theta_{[j]} - \theta_{[i]}\right)^2\right)^{1/2}$$

where $t = g+1, ..., G$

$r_i = arg\, min_{g+1<t<G}\left[RSSE(t,\, i)\right]$

5. $r = Mean(r_1, r_2, ..., r_g)$, the cutoff.

Algorithm 4.4 chooses the cutoff r by minimizing the RSSE for $\boldsymbol{\theta}^v = \left[\theta_{[1]}, ..., \theta_{[g]}\right]$. With the proposed choice of r, we propose to use $\left[\theta_{[1]}, ..., \theta_{[r]}\right]$ for ABC. The median can be used instead of the mean in Algorithm 4.4 at step 5, if the distribution of r is skewed, leading to the following version of the algorithm:

Algorithm 4.5 : choosing cutoff for ABC

1. Given a vector of parameter values $\boldsymbol{\theta}^* := \left[\theta_{[1]}, ..., \theta_{[N]}\right]$, sorted by Algorithm 4.1 according to $\rho(\mathbf{S}', \mathbf{S})$.
2. Specify search space for cutoff $r \in \{g+1, g+2, ..., G\}$, where g and G are lower bound and upper bound respectively. They are user define numbers.
3. Specify validation sample, $\boldsymbol{\theta}^{\boldsymbol{v}} = \left[\theta_{[1]}, ..., \theta_{[g]}\right]$
4. For $i = 1, ..., g$, repeat

   $RSSE\,(t,\, i) = \left(\frac{1}{t-g}\sum_{j=g+1}^{t}\left(\theta_{[j]} - \theta_{[i]}\right)^2\right)^{1/2}$

   where $t = g+1, ..., G$

   $r_i = arg\,min_{g+1<t<G}\left[RSSE(t,\, i)\right]$
5. $r = Median(r_1, r_2, ..., r_g)$, the cutoff.

The computational time of Algorithm 4.5 can be reduced by computing the $RSSE$ only once for $\bar{\theta} := \frac{\sum_{i=1}^{g}\theta_{[i]}^{v}}{g}$, where $\boldsymbol{\theta}^{\boldsymbol{v}} = \left[\theta_{[1]}, ..., \theta_{[g]}\right]$ be the sorted values of the parameter θ. This leads to our next algorithm given below:

Algorithm 4.6 : choosing cutoff for ABC

1. Given a vector of parameter values $\boldsymbol{\theta}^* := \left[\theta_{[1]}, ..., \theta_{[N]}\right]$, sorted by Algorithm 4.1 according to $\rho(\mathbf{S}', \mathbf{S})$.
2. Specify search space for cutoff $r \in \{g+1, g+2, ..., G\}$, where g and G are lower bound and upper bound respectively. They are user define numbers.
3. Specify validation sample, $\bar{\theta} := \frac{\sum_{i=1}^{g}\theta_{[i]}}{g}$,
4. Calculate $RSSE\,(t) = \left(\frac{1}{t-g}\sum_{j=g+1}^{t}\left(\theta_{[j]} - \bar{\theta}\right)^2\right)^{1/2}$, where $t = g+1, ..., G$.
5. $r = arg\,min_{g+1<t<G}\left(RSSE(t)\right)$, the cutoff.

In algorithm 4.6, $\bar{\theta}$ is used instead of the true parameter for validation. To explore search space of parameter efficiently, we introduce another user define number s (see

Algorithm 4.7).

Algorithm 4.7 : choosing cutoff for ABC

1. Given a vector of simulated parameter values $\boldsymbol{\theta}^* := \left[\theta_{[1]}, ..., \theta_{[N]}\right]$, sorted by Algorithm 4.1 according to $\rho(\mathbf{S}', \mathbf{S})$.
2. Specify search space for cutoff $r = \{g + 1, g + 2, ..., G\}$, where g and G are lower bound and upper bound respectively. They are user defined numbers.
3. Generate f uniformly distributed random numbers \$m_h\$ between 1 and s $h = 1, ..., f$. where f and s are user defined numbers, and s is the size of the validation sample.
4. Specify validation sample, $\bar{\theta}_{m_h} := \frac{\sum_{i=1}^{m_l} \theta^v_{[i]}}{m_h}$, $h = 1, ..., f$.
5. Calculate $RSSE(t, m_h) = \left(\frac{1}{t-g} \sum_{j=g+1}^{t} \left(\theta_{[j]} - \bar{\theta}_{m_h}\right)^2\right)^{1/2}$ where $t = g + 1, ..., G$.
6. Define $min_t = min_{g+1<m_h<s} (RSSE(t, m_h))$; $h = 1, ..., f$.
7. $r = arg\, min_{g+1<t<G} (RSSE(t))$, the cutoff.

The Algorithm 4.7 is quite similar to previous Algorithm 4.6 until step 3. An idea behind this algorithm is to minimize the rooted sum of square error (RSSE) of the mean of posterior distribution at random samples without replacement points between g and s. Where s is the search space for validation sample that have to specify in advance (see Table 4.7). We can also say that the Algorithm 4.6 is a special case of the Algorithm 4.7, because in the Algorithm 4.6 we take the mean of $\theta_{[1]}, ..., \theta_{[g]}$, but in the Algorithm 4.7 random f means between g and s are taken in account. Choice of G is depend on the algorithm (e.g., ABC-REJ, ABC-MCMC, ABC-PMC), and we could suggest from our simulation $G = 0.02 \times N$ that is a good choice for the ABC-REJ algorithm. An other user define parameter is g, and it is tradeoff between accuracy and number of accepted samples (see results in Table 4.6 and 4.7).

4.4.1 Simulation Results

Our simulation study is about the estimation of the scaled mutation and recombination rates, θ and ρ respectively, and the simulated data sets consist of 50 haplotypes being generated using the *ms* software [*Hudson*, 2002] under the standard neutral infinite-sites (IS) coalescent model *Nordborg* [2007]. The prior distribution is the following for the scaled mutation rate $\theta \sim U(0, 10)$ and for recombination $\rho \sim U(0, 10)$. The simulated and prior distribution are chosen same. The seven summary statistics have been calculated, and 100 observed data created under the true parameters $\theta = 7$, and $\rho = 7$. Here we compared the results of our proposed approach to the leave-one-out cross (LOOCV) validation procedure from the R "abc" package and the 1% acceptance cutoff (FIX01) method. We proceed with a simulation study in which $d = 10^2$ data sets were generated, and for each data set $N = 10^6$ ABC samples were simulated. The task is to choose which of the $N = 10^6$ simulated samples should be used for inference of the parameters θ and ρ with ABC (see Algorithm 4.1). The results are given below without regression adjustment. In this thesis, all simulation studies are using the Euclidean distances as distance metric ρ for ABC. The results are given below:

Table 4.6: Choice of g, in proposed algorithms.

Description		Mutation (θ)			Recombination (ρ)		
		Time	**MRSSE**	**Accept**	**Time**	**MRSSE**	**Accept**
$g = 20$	**Algorithm 4.4**	3.74	1.791	107	3.19	3.049	1086
	Algorithm 4.5	5.47	1.739	32	5.43	2.870	35
	Algorithm 4.6	0.14	1.713	32	0.14	2.844	34
	Algorithm 4.7	3.05	1.718	33	3.23	2.831	37
$g = 50$	**Algorithm 4.4**	7.42	1.817	830	7.43	3.107	2518
	Algorithm 4.5	13.33	1.772	85	13.27	2.965	71
	Algorithm 4.6	0.15	1.750	77	0.15	2.935	66
	Algorithm 4.7	3.05	1.755	84	3.23	2.931	69
$g = 100$	**Algorithm 4.4**	15.24	1.829	1558	14.53	3.118	3785
	Algorithm 4.5	26.33	1.791	140	26.11	2.997	135
	Algorithm 4.6	0.15	1.775	140	0.15	2.982	134
	Algorithm 4.7	3.05	1.775	142	3.23	2.982	136

On the basis of the results in Table 4.6, we would suggest that $g = 50$ is a reasonable choice for the algorithm with respect to both the number of accepted samples and computational time. For other problems, we suggest to do a similar calibration based on simulations to choose g. All the results in Table 4.6 are averages over 100 data sets, and also the computation time (in Hours) is for 100 data sets. Algorithm 4.5 performs better than Algorithm 4.4, probably because of skewness. We can see that Algorithm 4.6 is better and also faster than all the other methods. Algorithm 4.7 is slower but achieves a similar level of accuracy.

Table 4.7: MRSSE with respect to different user define values of g, f, s, with fixed $G = 0.02 * N$.

Description		**Mutation** (θ)					**Recombination** (ρ)				
$s \mid f$		5	10	20	50	100	5	10	20	50	100
$g = 20$	**200**	1.716	1.716	*1.718*	*1.718*	*1.718*	2.833	2.829	*2.832*	*2.832*	*2.832*
	500	1.718	1.717	*1.718*	*1.718*	*1.718*	2.837	2.831	*2.831*	*2.831*	*2.831*
	1000	1.718	1.717	*1.718*	*1.720*	*1.718*	2.823	2.835	*2.831*	*2.832*	*2.832*
	5000	1.718	1.715	1.721	1.714	1.716	2.835	2.836	2.832	2.836	2.831
	10000	1.719	1.719	1.716	1.716	1.716	2.827	2.827	2.825	2.834	2.833
	20000	1.719	1.706	1.713	1.719	1.714	2.819	2.809	2.829	2.821	2.828
$g = 50$	**200**	1.757	1.756	*1.754*	*1.755*	*1.755*	2.931	2.930	*2.931*	*2.931*	*2.931*
	500	1.756	1.759	*1.754*	*1.755*	*1.755*	2.929	2.928	*2.934*	*2.931*	*2.931*
	1000	1.757	1.757	*1.755*	*1.755*	*1.754*	2.929	2.931	*2.928*	*2.931*	*2.931*
	5000	1.758	1.756	1.758	1.756	1.755	2.931	2.926	2.924	2.926	2.926
	10000	1.756	1.756	1.756	1.758	1.759	2.928	2.929	2.935	2.921	2.922
	20000	1.756	1.756	1.757	1.757	1.758	2.923	2.922	2.920	2.920	2.928
$g = 100$	**200**	1.776	1.776	1.776	1.776	1.776	2.984	2.983	2.983	2.983	2.983
	500	1.775	1.776	1.776	1.776	1.776	2.980	2.983	2.983	2.983	2.983
	1000	1.775	1.776	1.776	1.775	1.776	2.983	2.979	2.981	2.983	2.983
	5000	1.777	1.777	1.775	1.775	1.776	2.982	2.980	2.982	2.983	2.982
	10000	1.778	1.774	1.774	1.776	1.776	2.988	2.982	2.979	2.980	2.984
	20000	1.776	1.776	1.776	1.775	1.774	2.985	2.982	2.980	2.981	2.983

From Table 4.7, we suggest that $g = 50$, $f \geq 20$, and a search space $s < 1000$ are reasonable choices. Moreover, the choice of g is more crucial than that of the other parameters f and s. These results are averages over 100 data sets, and with G fixed to $G = 0.02 * N$. As suggested by these results, we chose $g = 50$, $G = 0.2 * N$ and for Algorithm 4.7 $f = 50$, and $s = 500$ for each parameter. for the simulations leading to Table 4.8 and 4.9,

Table 4.8: Quantile of accepted observations by different algorithms.

Description	**Quantile (Mutation)**			**Quantile (Recombination)**		
	5%	**50%**	**95%**	**5%**	**50%**	**95%**
Optimum	52	118	15432	53	86	16876
Algorithm 4.4	70	830	5632	77	2518	7519
Algorithm 4.5	52	85	3913	53	71	1831
Algorithm 4.6	53	77	3717	53	66	11231
Algorithm 4.7	54	84	3672	52	69	11389
LOOCV	100	100	1000	100	100	1000

Table 4.8 shows the quantiles of the number accepted observations (out of 10^6) in 100 data sets. We can say that both methods selected approximately 100 observations on average (i.e., Median). For LOOCV, we used 50 cross-validation samples and three proposed acceptance cutoffs (i.e., 0.0001, 0.0005, 0.001).

Table 4.9: Performance of different algorithms by MRSSE.

Description	**Mutation (θ)**			**Recombination (ρ)**		
	Estimate $\hat{\theta}$		**MRSSE**	**Estimate $\hat{\rho}$**		**MRSSE**
	Mean	**S.E.**	$(\hat{\theta} - \theta)$	**Mean**	**S.E.**	$(\hat{\rho} - \rho)$
$\hat{\theta}_W$, $\hat{\rho}_{Hud}$	7.017	2.084	–	9.631	5.828	–
Optimum	6.967	1.076	1.698	5.851	0.958	2.831
Algorithm 4.4	6.939	1.210	1.817	5.487	0.815	3.107
Algorithm 4.5	6.989	1.254	1.772	5.619	1.268	2.965
Algorithm 4.6	6.976	1.236	**1.750**	5.618	1.215	2.935
Algorithm 4.7	6.987	1.244	1.755	5.622	1.212	**2.931**
LOOCV	6.946	1.203	1.827	5.534	1.088	3.088
FIX01	6.899	1.191	1.875	5.425	0.715	3.187

In the above table, θ_W is the Watterson estimator [*Watterson*, 1975], and ρ_{Hud} is Hudson's estimator of recombination [*Hudson*, 2001]. Table 4.9 results show that the Algorithm 4.6 and 4.7 give smaller MRSSE than all other methods, and these results are close to those obtained under the idealized situation when using the algorithms with the true parameter values for computing an optimum MRSSE.

This page intentionally left blank

Chapter 5

Summary and Conclusions

In the past decade, remarkable advances have been made in the field of biology. Nowadays, biologists who study natural populations of plants and animals, have access to numerous new tools such as whole genome sequencing, DNA hybridization microarrays, and next-generation sequencing. Computationally intensive statistical methods have to be developed often for the analysis of complicated biological data. Of course, the advancement in the field of computing has been equally significant, and today's computers are fast enough to allow numerically intensive analysis to be run on desktop machines. This has led to a substantial progress in developing statistical methods for genetics; in particular, Markov chain Monte Carlo (MCMC) and Approximate Bayesian Computation (ABC) methods for computing likelihoods and posterior probabilities. The main objective of this study is to deal with statistical challenges in modern genetics. Both likelihood and likelihood-free methods are needed for the analysis of genetic data in the context of questions of interest to biologists. In this thesis, we follow both aggressively. We proposed a novel method for the estimation of time dependent scaled mutation rates under the infinite sites model when recombination is not present. The proposed method can also estimate time-independent mutation rates, and it performs well than the methods in the literature. Second, on the likelihood-free idea, we investigate a method for choosing summary

statistics in ABC algorithm, and it performs better in terms of computational time and accuracy than the methods given in the literature. Moreover, four new algorithms have been proposed for choosing the acceptance cutoff in ABC framework.

5.1 Exact Likelihood Calculation

A novel method is proposed that is based on dynamic programming for estimating the scaled mutation rate θ that dependent on time t, under the infinite sites model in absent of recombination. The proposed method efficiently computes the probabilities of all possible configurations at each step, i.e., coalescence or mutation at a reasonable time, although intermediate configuration are large. The results of the proposed method are reasonable for growing population (see Table 3.2). When number of independent loci increases the amount of outlier's decreases (see Figure 3.1) and estimation of parameters become precise to attain its true parameter. Contour plots also show that our proposed method successfully estimate three parameters such that the current theta ($\hat{\theta}_C$), the ancestral theta ($\hat{\theta}_A$) and the time ($\hat{\tau}$), when population is growing.

With a little modification, the proposed method also estimate the scaled mutation rate θ that is independent of time. The performance of the proposed method is compared to Griffiths and Tavaré (GT) approach that is implemented in the Genetree program (see Table 3.1). The results show that the proposed method is fast and reliable when $n\theta < 100$, where θ is the time-independent mutation and n is sample size.

5.2 Choosing Summary Statistics for ABC

We have investigated a method to select a set of informative summary statistics for the parameter $\boldsymbol{\theta}$ in an ABC framework. We found that the LARS algorithm is super fast and producing relatively better results than the other methods such as PLS, AS, ME. (see Example 1, Section 4.3.1). The results of the two-stage (2S) algorithm are slightly better. However, the 2S algorithm is computationally very intensive. The two stage (2S) algorithm took 12 hours in Example 1 with seven summary statistics only. On the other hand, the computational time for LARS is not more than 5 minutes, and is therefore much faster than the two-stage (2S) algorithm. In example 1, we are expecting that C1 (number of segregating sites) for θ [*Hudson*, 1990; *Nordborg*, 2007] and C5 (number of distinct haplotype) for ρ are particularly informative summary statistics, and they should be included in an optimal set of summary statistics.

In example 2 (see Section 4.3.2), we compared the LARS results with the PLS method because other methods are computationally too intensive. Selected informative summary statistics by LARS are quite similar as we are expecting [see for description of summary statistics *Pavlidis et al.*, 2010].

5.3 Choosing Acceptance Cutoff for ABC

We proposed four new algorithms and compared their results based on our simulation study (see Section 4.4.1). We could suggest that the algorithm 4.6 is better than the other considered approaches in terms of computational time and accuracy of the estimates. We also tried the super learner (SL) approach for choosing the acceptance cutoff in an ABC framework, but it turned out to be computationally too expensive.

5.4 Future Recommendations

In this thesis, we tried to develop reliable methods for biological data. A novel method has been developed based on likelihood inference for the estimation of the time-dependent mutation rate. This method could be further extended to estimate migration rates. We also developed an algorithm for choosing summary statistics and algorithms for the selection of the acceptance cutoff for ABC. We hope to have contributed to the development of improved methods to understand the variation pattern in evolution and in genetics.

Bibliography

Beaumont, M. A., Approximate Bayesian in Evolution and Ecology, *Annual review of ecology, evolution, and systematics*, *41*, 379–406, 2010.

Beaumont, M. A., W. Zhang, and D. J. Balding, Approximate Bayesian Computation in Population Genetics, *Genetics*, *162*(4), 2025–2035, 2002.

Beaumont, M. A., J.-M. Cornuet, J.-M. Marin, and C. P. Robert, Adaptive approximate Bayesian computation, *Biometrika*, *96*(4), 983–990, doi:10.1093/biomet/asp052, 2009.

Bertorelle, G., A. Benazzo, and S. Mona, ABC as a flexible framework to estimate demography over space and time: some cons, many pros., *Molecular ecology*, *19*(13), 2609–2625, doi:10.1111/j.1365-294X.2010.04690.x, 2010.

Besag, J., Spatial interaction and the statistical analysis of lattice systems, *Journal of the Royal Statistical Society, Series B*, *36*(2), 192 – 236, 1974.

Beven, K., A manifesto for the equifinality thesis, *Journal of hydrology*, *320*(1-2), 18–36, 2006.

Blum, M. G. B., Approximate Bayesian Computation: A Nonparametric Perspective, *Journal of the American Statistical Association*, *105*(491), 1178–1187, doi:10.1198/jasa.2010.tm09448, 2010.

Blum, M. G. B., and O. François, Non-linear regression models for Approximate Bayesian Computation, *Statistics and Computing*, *20*(1), 63–73, doi:10.1007/s11222-009-9116-0, 2009.

Blum, M. G. B., and V. C. Tran, HIV with contact tracing: a case study in approximate Bayesian computation., *Biostatistics (Oxford, England)*, *11*(4), 644–660, doi:10.1093/biostatistics/kxq022, 2010.

Bortot, P., S. G. Coles, and S. A. Sisson, Inference for Stereological Extremes, *Journal of the American Statistical Association*, *102*(477), 84–92, doi:10.1198/016214506000000988, 2007.

Breiman, L., and P. Spector, Submodel selection and evaluation in regression. The X-random case, *International statistical review*, *60*(3), 291–319, 1992.

Breiman, L., J. Friedman, J. C. Stone, and R. Olshen, *Classification and regression trees*, 1st ed., 1–358 pp., Wadsworth International Group, Wadsworth, 1984.

Bürger, R., *The mathematical theory of selection recombination and mutation*, 409 pp., John Wiley, 2000.

Cam, L. L., Sufficiency and Approximate Sufficiency, *The Annals of Mathematical Statistics*, *35*(4), 1419–1455, 1964.

Cappé, O., A. Guillin, J. M. Marin, and C. P. Robert, Population Monte Carlo, *Journal of Computational and Graphical Statistics*, *13*(4), 907–929, doi:10.1198/106186004X12803, 2004.

Cohen, R., Introducing the GLMSELECT Procedure for Model Selection, in *Proceedings of the Thirty-First Annual SAS Users Group International Conference*, 2006.

Cornuet, J. M., J. M. Marin, A. Mira, and C. P. Robert, Adaptive Multiple Importance Sampling, *HAL : hal-00403248,version 1*, 2009.

Csilléry, K., M. G. B. Blum, O. E. Gaggiotti, and O. François, Approximate Bayesian Computation (ABC) in practice, *Trends in Ecology & Evolution*, *25*(7), 410–418, doi:10.1016/j.tree.2010.04.001, 2010.

Del Moral, P., A. Doucet, and A. Jasra, Sequential Monte Carlo samplers, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *68*(3), 411–436, doi:10.1111/j.1467-9868.2006.00553.x, 2006.

Del Moral, P., A. Doucet, and A. Jasra, On Adaptive Resampling Procedures for Sequential Monte Carlo Methods, *inria-00332436, version 4*, 2010.

Donnelly, P., and S. Tavaré, Coalescents and genealogical structure under neutrality, *Annual review of genetics*, *29*, 401–421, doi:10.1146/annurev.ge.29.120195.002153, 1995.

Donnelly, P. J., and S. Tavaré, *Progress in population genetics and human evolution*, 329 pp., Springer, New York, 1997.

Drovandi, C. C., and A. N. Pettitt, Estimation of Parameters for Macroparasite Population Evolution Using Approximate Bayesian Computation., *Biometrics DOI: 10.1111/j.1541-0420.2010.01410.x*, doi:10.1111/j.1541-0420.2010.01410.x, 2010.

Efron, B., T. Hastie, I. Johnstone, and R. Tibshirani, Least Angle Regression, *The Annals of Statistics*, *32*(2), 407 – 451, 2004.

Estoup, A., P. Jarne, and J. M. Cornuet, Homoplasy and mutation model at microsatellite loci and their consequences for population genetics analysis, *Molecular Ecology*, *11*(9), 1591–1604, doi:10.1046/j.1365-294X.2002.01576.x, 2002.

Ewens, W., Population Genetics Theory - The Past and the Future, pp. 177 – 227, 1990.

Ewens, W. J., The sampling theory of selectively neutral alleles, *Theoretical Population Biology*, *3*(1), 87–112, doi:10.1016/0040-5809(72)90035-4, 1972.

Ewens, W. J., *Mathematical Population Genetics: Theoretical Introduction*, 2nd ed., 1–417 pp., Springer, 2004.

Excoffier, L., and G. Heckel, Computer programs for population genetics data analysis: a survival guide., *Nature reviews. Genetics*, *7*(10), 745–758, doi:10.1038/nrg1904, 2006.

Fabrice, L., and F. Paul, On Composite Likelihoods in Statistical Genetics, *Statistica Sinica*, *21*, 43–69, 2011.

Fagundes, N. J. R., N. Ray, M. Beaumont, S. Neuenschwander, F. M. Salzano, S. L. Bonatto, and L. Excoffier, Statistical evaluation of alternative models of human evolution., *Proceedings of the National Academy of Sciences of the United States of America*, *104*(45), 17,614–17,619, doi:10.1073/pnas.0708280104, 2007.

Fay, J. C., and C.-I. Wu, Hitchhiking Under Positive Darwinian Selection, *Genetics*, *155*(3), 1405–1413, 2000.

Fearnhead, P., and D. Prangle, Semi-automatic Approximate Bayesian Computation, *Arxiv preprint arXiv:1004.1112.*, 2010.

Felsenstein, J., M. K. Kuhner, J. Yamato, and P. Beerli, Likelihoods on Coalescents: A Monte Carlo Sampling Approach to Inferring Parameters from Population Samples of Molecular Data, *Statistics in Molecular Biology, IMS Lecture Notes-Monograph Series*, *33*(1), 163–185, 1999.

Fisher, R. A., *The Genetical Theory of Natural Selection*, Clarendon Press, Oxford, UK, 1930.

Fu, Y. X., and W. H. Li, Statistical tests of neutrality of mutations, *Genetics*, *133*(3), 693–709, 1993a.

Fu, Y. X., and W. H. Li, Maximum likelihood estimation of population parameters, *Genetics*, *134*(4), 1261–1270, 1993b.

Fu, Y. X., and W. H. Li, Estimating the age of the common ancestor of a sample of DNA sequences, *Mol. Biol. Evol.*, *14*(2), 195–199, 1997.

Futschik, A., and F. Gach, On the inadmissibility of Watterson's estimator., *Theoretical population biology*, *73*(2), 212–221, doi:10.1016/j.tpb.2007.11.009, 2008.

Gonzalez-Pastor, J., J. San Millan, M. Castilla, and F. Moreno, Structure and organization of plasmid genes required to produce the translation inhibitor microcin C7, *J. Bacteriol.*, *177*(24), 7131–7140, 1995.

Gourieroux, C., A. Monfort, and E. Renault, Indirect inference, *Journal of Applied Econometrics*, *8*(S1), S85–S118, doi:10.1002/jae.3950080507, 1993.

Griffiths, R., and P. Marjoram, Ancestral Inference from Samples of DNA Sequences with Recombination, *Journal of Computational Biology*, *3*(4), 479–502, doi:10.1089/cmb.1996.3.479, 1996.

Griffiths, R., and S. Tavaré, Computational methods for the coalescent. In: Donnelly, P. and Tavaré, S. (Eds.), Progress in Population Genetics and Human Evolution, IMA Volumes in Mathematics and its Applications,, *Springer Verlag, Berlin*, *87*, 165–182, 1997.

Griffiths, R. C., Lines of descent in the diffusion approximation of neutral Wright-Fisher models, *Theoretical Population Biology*, *17*(1), 37–50, doi:10.1016/0040-5809(80)90013-1, 1980.

Griffiths, R. C., and S. Tavaré, Simulating Probability Distributions in the Coalescent, *Theoretical Population Biology*, *46*(2), 131–159, doi:10.1006/tpbi.1994.1023, 1994a.

Griffiths, R. C., and S. Tavaré, Ancestral Inference in Population Genetics, *Statistical Science*, *9*(3), 307 – 319, 1994b.

Grimm, V., et al., Pattern-oriented modeling of agent-based complex systems: lessons from ecology., *Science (New York, N.Y.)*, *310*(5750), 987–91, doi:10.1126/science.1116681, 2005.

Haldane, J., *The causes of natural selection*, Longmans Green, London, 1932.

Hamilton, G., M. Currat, N. Ray, G. Heckel, M. Beaumont, and L. Excoffier, Bayesian estimation of recent migration rates after a spatial expansion., *Genetics*, *170*(1), 409–417, doi:10.1534/genetics.104.034199, 2005.

Hardy, G. H., Mendelian Proportions in a Mixed Population., *Science (New York, N.Y.)*, *28*(706), 49–50, doi:10.1126/science.28.706.49, 1908.

Harris, H., Enzyme Polymorphisms in Man, *Proceedings of the Royal Society of London. Series B, Biological Sciences*, *164*(995), 298 – 310, 1966.

Hastie, T., R. Tibshirani, and H. J. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, 2nd ed., 1–732 pp., Springer Science Business Media, LLC, New York, 2009.

Hein, J., M. H. Schierup, and C. Wiuf, *Gene genealogies, variation and evolution: a primer in coalescent theory*, Oxford University Press, 2005.

Hobolth, A., M. K. Uyenoyama, and C. Wiuf, Importance sampling for the infinite sites model., *Statistical applications in genetics and molecular biology*, *7*(1), Article32, doi:10.2202/1544-6115.1400, 2008.

Hudson, R., D. Boos, and N. Kaplan, A statistical test for detecting geographic subdivision, *Mol. Biol. Evol.*, *9*(1), 138–151, 1992a.

Hudson, R. R., Testing the Constant-Rate Neutral Allele Model with Protein Sequence Data, *Evolution*, *37*(1), 203 – 217, 1983.

Hudson, R. R., Gene genealogies and the coalescent process, *Oxford Survey Evol. Biol.*, *7*(1), 1–44, 1990.

Hudson, R. R., Two-Locus Sampling Distributions and Their Application, *Genetics*, *159*(4), 1805–1817, 2001.

Hudson, R. R., Generating samples under a Wright-Fisher neutral model of genetic variation., *Bioinformatics (Oxford, England)*, *18*(2), 337–338, 2002.

Hudson, R. R., M. Slatkin, and W. P. Maddison, Estimation of Levels of Gene Flow From DNA Sequence Data, *Genetics*, *132*(2), 583–589, 1992b.

Hyrien, O., M. Mayer-Pröschel, M. Noble, and A. Yakovlev, A stochastic model to analyze clonal data on multi-type cell populations., *Biometrics*, *61*(1), 199–207, doi:10.1111/j.0006-341X.2005.031210.x, 2005.

Jabot, F., and J. Chave, Inferring the parameters of the neutral theory of biodiversity using phylogenetic information and implications for tropical forests., *Ecology letters*, *12*(3), 239–248, doi:10.1111/j.1461-0248.2008.01280.x, 2009.

Joyce, P., and P. Marjoram, Approximately sufficient statistics and bayesian computation., *Statistical applications in genetics and molecular biology*, *7*(1), Article 26, doi:10.2202/1544-6115.1389, 2008.

Kelly, J. K., A test of neutrality based on interlocus associations., *Genetics*, *146*(3), 1197–206, 1997.

Kimura, M., "Stepping Stone" model of population., *Ann. Rept. Nat. Inst. Genetics Japan*, *3*, 62–63, 1953.

Kimura, M., Stochastic processes and distribution of gene frequencies under natural selection., *Cold Spring Harbor symposia on quantitative biology*, *20*, 33–53, 1955.

Kimura, M., The number of heterozygous nucleotide sites maintained in a finite population due to steady flux of mutations., *Genetics*, *61*(4), 893–903, 1969.

Kimura, M., and J. F. Crow, The number of alleles that can be maintained in a finite population., *Genetics*, *49*, 725–738, 1964.

Kimura, M., and T. Ohta, The age of a neutral mutant persisting in a finite population, *Genetics*, *75*(1), 199–212, 1973.

Kingman, J. F., Origins of the coalescent. 1974-1982, *Genetics*, *156*(4), 1461–1463, 2000.

Kingman, J. F. C., Exchangeability and the evolution of large populations, *Exchangeability in Probability and Statistics edited by G. Koch and F. Spizzichino. North-Holland, Amsterdam.*, pp. 97–112, 1982a.

Kingman, J. F. C., The coalescent, *Stochastic Processes and their Applications*, *13*(3), 235–248, doi:10.1016/0304-4149(82)90011-4, 1982b.

Kingman, J. F. C., On the Genealogy of Large Populations, *Journal of Applied Probability*, *19A*, 27– 43, 1982c.

Kohavi, R., A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection, in *International Joint Conference on Arti cial Intelligence*, pp. 1137 – 1145, 1995.

Kuhner, M. K., J. Yamato, and J. Felsenstein, Estimating effective population size and mutation rate from sequence data using Metropolis-Hastings sampling., *Genetics*, *140*(4), 1421–1430, 1995.

Lane, R. O., M. Briers, and K. Copsey, Approximate Bayesian Computation for Source Term Estimation, 2009.

Leuenberger, C., and D. Wegmann, Bayesian computation and model selection without likelihoods., *Genetics*, *184*(1), 243–252, doi:10.1534/genetics.109.109058, 2010.

Lewontin, R. C., and J. L. Hubby, A molecular approach to the study of genic heterozygosity in natural populations. II. Amount of variation and degree of heterozygosity in natural populations of Drosophila pseudoobscura., *Genetics*, *54*(2), 595–609, 1966.

Li, B., and P. McCullagh, Potential functions and conservative estimating functions, *Annals of statistics*, *22*(1), 340–356, 1994.

Lindsay, B. G., Composite likelihood methods. , *Contemporary Mathematics*, *80*, 221–239, 1988.

Lopes, J. S., and M. A. Beaumont, ABC: A useful Bayesian tool for the analysis of population data., *journal of molecular epidemiology and evolutionary genetics in infectious diseases*, *10*(6), 825–832, doi:10.1016/j.meegid.2009.10.010, 2010.

Luciani, F., S. A. Sisson, H. Jiang, A. R. Francis, and M. M. Tanaka, The epidemiological fitness cost of drug resistance in Mycobacterium tuberculosis., *Proceedings of the National Academy of Sciences of the United States of America*, *106*(34), 14,711–14,715, doi:10.1073/pnas.0902437106, 2009.

Malècot, G., La consanguinity dans une population limitèe. , *C. R. Acad. Sci. Paris*, *222*, 841–843, 1946.

Malécot, G., *Les mathématiques de l'hérédité*, Masson, 1948.

Marjoram, P., and S. Tavaré, Modern computational approaches for analysing molecular genetic variation data., *Nature reviews. Genetics*, *7*(10), 759–770, doi:10.1038/nrg1961, 2006.

Marjoram, P., J. Molitor, V. Plagnol, and S. Tavare, Markov chain Monte Carlo without likelihoods., *Proceedings of the National Academy of Sciences of the United States of America*, *100*(26), 15,324–15,328, doi:10.1073/pnas.0306899100, 2003.

McCullagh, P., and J. A. Nelder, *Generalized Linear Models.*, 2nd ed., Chapman and Hall,, London, 1989.

McVean, G., P. Awadalla, and P. Fearnhead, A Coalescent-Based Method for Detecting and Estimating Recombination From Gene Sequences, *Genetics*, *160*(3), 1231–1241, 2002.

Mendel, G., Versuche über Pflanzenhybriden Verhandlungen des naturforschenden Vereines in Brünn. Translated by Druery, C.T and William Bateson in 1901., *Abhandlungen*, *4*, 3–47, 1866.

Mevik, B. H., and R. Wehrens, The pls Package: Principal Component and Partial Least Squares Regression in R, *Journal of Statistical Software*, *18*(2), 2007.

Möhle, M., Ancestral processes in population genetics-the coalescent., *Journal of theoretical biology*, *204*(4), 629–638, doi:10.1006/jtbi.2000.2032, 2000.

Moran, P. A. P., Random processes in genetics, *Mathematical Proceedings of the Cambridge Philosophical Society*, *54*(01), 60–71, 1958.

Moran, P. A. P., *The statistical processes of evolutionary theory*, Clarendon Press, Oxford, 1962.

Moran, P. A. P., The estimation of standard errors in Monte Carlo simulation experiments, *Biometrika*, *62*(1), 1–4, doi:10.1093/biomet/62.1.1, 1975.

Nagylaki, T., Rate of evolution of a character without epistasis., *Proceedings of the National Academy of Sciences of the United States of America*, *86*(6), 1910–1913, 1989.

Nevat, I., G. W. Peters, A. Doucet, and J. Yuan, Channel Tracking for Relay Networks via Adaptive Particle MCMC, *Arxiv*, 2010.

Nielsen, R., Maximum likelihood estimation of population divergence times and population phylogenies under the infinite sites model, *Theoretical population biology*, *53*(2), 143–151, doi:10.1006/tpbi.1997.1348, 1998.

Nordborg, M., Coalescent theory, pp. 179 -212 in Handbook of Statistical Genetics, edited by D. J. BALDING, M. J. BISHOP and C. CANNINGS. John Wiley & Sons, Chichester, UK., 2001.

Nordborg, M., Coalescent theory , in *Handbook of Statistical Genetics*, 3rd ed., pp. 179–208, Wiley: Chichester, 2007.

Nunes, M. A., and D. J. Balding, On Optimal Selection of Summary Statistics for Approximate Bayesian Computation, *Statistical Applications in Genetics and Molecular Biology*, *9*(1), Article 34, 2010.

Pavlidis, P., S. Laurent, and W. Stephan, msABC: a modification of Hudson's ms to facilitate multi-locus ABC analysis, *Molecular Ecology Resources*, *10*(4), 723–727, doi:10.1111/j.1755-0998.2010.02832.x, 2010.

Peters, G. W., and S. A. Sisson, Bayesian inference, Monte Carlo sampling and operational risk, *Journal of Operational Risk,*, *1*(3), 2006.

Peters, G. W., Y. Fan, and S. A. Sisson, On sequential Monte Carlo, partial rejection control and approximate Bayesian computation, *Tech. rep.*, Technical report, University of New South Wales., Wales, 2008.

Peters, G. W., S. A. Sisson, and Y. Fan, Likelihood-free Bayesian models for -stable models, *Tech. rep.*, Technical report, University of New South Wales., Wales, 2009.

Pinheiro, H. P., S. F. Kiihl, A. Pinheiro, and S. F. dos Reis, Asymptotic behavior of the scaled mutation rate estimators., *Biometrical journal. Biometrische Zeitschrift*, *52*(3), 400–416, doi:10.1002/bimj.200900014, 2010.

Press, W., S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes: The Art of Scientific Computing*, third ed., 502–507 pp., Cambridge University Press, New York, 2007.

Provine, W. B., *The origins of theoretical population genetics*, University of Chicago Press, Chicago and London, 1971.

Ratmann, O., O. Jø rgensen, T. Hinkley, M. Stumpf, S. Richardson, and C. Wiuf, Using Likelihood-Free Inference to Compare Evolutionary Dynamics of the Protein Networks of H. pylori and P. falciparum, *PLoS Computational Biology*, *3*, e230, 2007.

Ratmann, O., C. Andrieu, C. Wiuf, and S. Richardson, Model criticism based on likelihood-free inference, with an application to protein network evolution., *Proceedings of the National Academy of Sciences of the United States of America*, *106*(26), 10,576–10,581, doi:10.1073/pnas.0807882106, 2009.

Reid, N., Likelihood inference, *Wiley Interdisciplinary Reviews: Computational Statistics*, *2*(5), 517–525, doi:10.1002/wics.110, 2010.

Ripley, B. D. (Ed.), *Stochastic Simulation*, Wiley Series in Probability and Statistics, John Wiley & Sons, Inc., Hoboken, NJ, USA, doi:10.1002/9780470316726, 1987.

Sisson, S. A., Y. Fan, and M. M. Tanaka, Sequential Monte Carlo without likelihoods., *Proceedings of the National Academy of Sciences of the United States of America*, *104*(6), 1760–1765, doi:10.1073/pnas.0607208104, 2007.

Slatkin, M., Isolation by Distance in Equilibrium and Non-Equilibrium Populations, *Evolution*, *47*(1), 264, doi:10.2307/2410134, 1993.

Slatkin, M., and M. Veuille, *Modern developments in theoretical population genetics: the legacy of Gustave Malècot*, Oxford University Press, 2002.

Sousa, V. C., M. Fritz, M. A. Beaumont, and L. Chikhi, Approximate bayesian computation without summary statistics: the case of admixture., *Genetics*, *181*(4), 1507–1519, doi:10.1534/genetics.108.098129, 2009.

Stephens, M., Problems with computational methods in population genetics, *Bulletin of the 52nd Session of the International Statistical Institute*, (1), 273–276, 1999.

Stephens, M., and P. Donnelly, Inference in molecular population genetics, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *62*(4), 605–635, doi:10.1111/1467-9868.00254, 2000.

Tajima, F., Evolutionary relationship of DNA sequences in finite populations, *Genetics*, *105*(2), 437–460, 1983.

Tajima, F., Statistical method for testing the neutral mutation hypothesis by DNA polymorphism, *Genetics*, *123*(3), 585–595, 1989.

Tajima, F., The Amount of DNA Polymorphism Maintained in a Finite Population When the Neutral Mutation Rate Varies Among Sites, *Genetics*, *143*(3), 1457–1465, 1996.

Tanaka, M. M., A. R. Francis, F. Luciani, and S. A. Sisson, Using approximate Bayesian computation to estimate tuberculosis transmission parameters from genotype data., *Genetics*, *173*(3), 1511–1520, doi:10.1534/genetics.106.055574, 2006.

Tavaré, S., Line-of-descent and genealogical processes, and their applications in population genetics models, *Theoretical Population Biology*, *26*(2), 119–164, doi:10.1016/0040-5809(84)90027-3, 1984.

Tavaré, S., D. J. Balding, R. C. Griffiths, and P. Donnelly, Inferring Coalescence Times From DNA Sequence Data, *Genetics*, *145*(2), 505–518, 1997.

Tishkoff, S. A., et al., Haplotype diversity and linkage disequilibrium at human G6PD: recent origin of alleles that confer malarial resistance., *Science (New York, N.Y.)*, *293*(5529), 455–462, doi:10.1126/science.1061573, 2001.

Toni, T., D. Welch, N. Strelkowa, A. Ipsen, and M. P. H. Stumpf, Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems, *Journal of The Royal Society Interface*, *6*(31), 187–202, doi:10.1098/rsif.2008.0172, 2009.

Varin, C., N. Reid, and D. Firth, An overview of composite likelihood methods., *Statistica Sinica*, *21*(1), 5–42, 2011.

Wakeley, J., Recent trends in population genetics: more data! More math! Simple models?, *The Journal of heredity*, *95*(5), 397–405, doi:10.1093/jhered/esh062, 2004.

Wakeley, J., *Coalescent theory: an introduction*, Roberts & Co. Publishers, 2009.

Wall, M. A., M. Socolich, and R. Ranganathan, The structural basis for red fluorescence in the tetrameric GFP homolog DsRed., *Nature structural biology*, *7*(12), 1133–1138, doi:10.1038/81992, 2000.

Watterson, G. A., On the number of segregating sites in genetical models without recombination, *Theoretical Population Biology*, *7*(2), 256–276, doi:10.1016/0040-5809(75)90020-9, 1975.

Wegmann, D., C. Leuenberger, and L. Excoffier, Efficient approximate Bayesian computation coupled with Markov chain Monte Carlo without likelihood., *Genetics*, *182*(4), 1207–1218, doi:10.1534/genetics.109.102509, 2009.

Weinberg, W., On the demonstration of heredity in man. Translated by Boyer SH IV (1963), *Papers on Human Genetics*, 1908.

Weiss, G., and A. von Haeseler, Inference of Population History Using a Likelihood Approach, *Genetics*, *149*(3), 1539–1546, 1998.

Wiegand, T., F. Jeltsch, I. Hanski, and V. Grimm, Using pattern-oriented modeling for revealing hidden information: a key for reconciling ecological theory and application, *Oikos*, *100*(2), 209–222, doi:10.1034/j.1600-0706.2003.12027.x, 2003.

Wiegand, T., E. Revilla, and F. Knauer, Dealing with Uncertainty in Spatially Explicit Population Models, *Biodiversity and Conservation*, *13*(1), 53–78, doi:10.1023/B:BIOC.0000004313.86836.ab, 2004.

Wilkinson, R. D., Approximate Bayesian computation (ABC) gives exact results under the assumption of model error, *Biometrika*, *20*(10), 1–13, 2008.

Wilkinson, R. D., and S. Tavaré, Estimating primate divergence times by using conditioned birth-and-death processes., *Theoretical population biology*, *75*(4), 278–285, doi:10.1016/j.tpb.2009.02.003, 2009.

Wood, S. N., Statistical inference for noisy nonlinear ecological dynamic systems, *Nature*, *466*(7310), 1102–1104, doi:10.1038/nature09319, 2010.

Wright, S., Evolution in Mendelian Populations, *Genetics*, *16*(2), 97–159, 1931.

Wright, S., Isolation by Distance., *Genetics*, *28*(2), 114–138, 1943.

Wu, Y., Exact Computation of Coalescent Likelihood under the Infinite Sites Model, in *ISBRA '09 Proceedings of the 5th International Symposium on Bioinformatics Research and Applications*, 2009.

Yule, G. U., Mendel's Laws and Their Probable Relations to Intra-Racial Heredity (Continued), *New Phytologist*, *1*(10), 222 – 238, 1902.

Appendix

Appendix I: Summary Statistics

Following is explanation of summary statistics from Example 1, which were presented in Table 4.1. Number of segregating sites (C1) – It is calculated just by counting the number of segregating sites in DNA sequence data. Uniform random variable (C2) -Its range is [0, 25]. Mean of pairwise difference (C3) – It is based on the average differences between two DNA sequences that were randomly chosen from sample. This index is commonly used in population genetics since its suggestion in 1979 by Nei and Li.

$$\pi = \frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{j=i+1}^{n-1} f_i f_j \pi_{ij},$$

where f_i and f_j are the frequencies of the i^{th} and j^{th} sequences, respectively, in the population, and π_{ij} is the number of nucleotide differences per nucleotide site between the i^{th} and j^{th} sequences. Linkage disequilibrium (C4)- $r^2 = (\frac{D}{p_1 p_2 q_1 q_2})^2$, where $D = x12 - p_1 p_2$. The following table illustrates the relationship between the haplotype frequencies ($x11$, $x12$, $x21$, $x22$) and allele frequencies ($A1$, $A2$, $B1$, $B2$) and D.

	A1	A2	Total
B1	$x11 = p_1q_1 + D$	$x21 = p_2q_1 - D$	q_1
B2	$x12 = p_1q_2 - D$	$x22 = p_2q_2 + D$	q_2
Total	p_1	p_2	1

And other three summary statistics (C5, C6, C7) are straight forward.

Example 2

Description	**List of all Summary Statistics**
C1	number of segregating sites for sub-population 1
C2	number of segregating sites for sub-population 2
C3	number of segregating sites for total sample
C4	the pi for sub-population 1
C5	the pi for sub-population 2
C6	the pi for total sample
C7	the Watterson estimator for sub-population 1
C8	the Watterson estimator for sub-population 2
C9	the Watterson estimator for total sample
C10	the Tajima's D for sub-population 1
C11	the Tajima's D for sub-population 2
C12	the Tajima's D for total sample
C13	the Zns for sub-population 1
C14	the Zns for sub-population 2
C15	the Zns for total sample
C16	the Fst (total sample, hbk calculation)
C17	the percentage of shared polymorphisms between sub-populations 1 and 2

C18	the percentage of private polymorphisms between sub-populations 1 and 2
C19	the percentage of fixed differences polymorphisms between sub-populations 1 and 2
C20	the Fst between sub-populations 1 and 2
C21	the H in sub-population 1
C22	the H in sub-population 2
C23	the H in total sample
C24	the number of haplotypes in sub-population 1
C25	the Heterozygosity of haplotypes in sub-population 1
C26	the number of haplotypes in sub-population 2
C27	the Heterozygosity of haplotypes in sub-population 2
C28	the number of haplotypes in the total sample
C29	the Heterozygosity of haplotypes in the total sample
C30	Uniform [0,1] random variable
C31	Uniform [0,10] random variable
C32	Uniform [0,25] random variable

For more description of the formulas these summary statistics[see *Pavlidis et al.*, 2010].

Appendix II: A C++ Program for Forward Algorithm

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

tree.h

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

```
#ifndef TREE
#define TREE
using namespace std;
#include<vector>
#include<iostream>
#include<string>
#include<assert.h>
#include<map>
#include<iterator>
/*
For a project to calculate the probability of a rooted coalescence tree
given theta.
*/
struct tree_error{
const char* pchar;
tree_error(const char* pch){pchar= pch;}
};
void seq2haplotype(const vector<vector<char> >& vvc,
int len,
map<vector<int>,int>& m_ht);
//a data vector of zero's and one's; and the length of the sequence in bps
typedef map<vector<int>,int> haplotype_map;
typedef map<haplotype_map,double> m_hm;
```

```
typedef haplotype_map::const_iterator CI;
typedef haplotype_map::iterator I;
typedef map<haplotype_map,double>::const_iterator CI_mm;
typedef map<haplotype_map,double>::iterator I_mm;
typedef map<haplotype_map,vector<double> > m_hmvd;
typedef map<haplotype_map,vector<double> >::const_iterator CI_mmvd;
typedef map<haplotype_map,vector<double> >::iterator I_mmvd;
void all_below(const haplotype_map& m_ht,
double prob, //the probability of the haplotype
double theta,
m_hm & map_of_htm);
void all_below_time(const haplotype_map& m_ht,
vector<double> vdprob, //the probability of the haplotype
double theta_start,
double theta_end,
double time,
double nu, // nu=1/(2Ne)
m_hmvd & map_of_htm);
#endif //TREE
```

```
######################
FORWARD_STEP.CC
######################

    #include<iostream>
    #include<fstream>
    #include<sstream>
    #include<vector>
    #include<string>
    #include"tree.h"
    #include<algorithm>
    #include<set>
    #include<math.h>
    struct named_sequence{
    string s; //name
    vector<char> seq; //sequence
    int length; //length of the sequence
    int operator < (const named_sequence& ns) const {return s<ns.s;}
    //order by name
    };
    void read_sequences(vector<named_sequence>& vns,
    ifstream& ifs)
    {//sequences must be in fasta format
    vns.resize(0);
    for(;;){
    char ach[10000];
    ifs.getline(ach,10000);
    if(!ifs) return;
    if(ach[0]!='>'){
    cerr << "Not in fasta format!" << endl;
```

```
exit(1);
}
named_sequence ns;
string s(&ach[1]);
istringstream istr(s);
char ach2[100];
istr.getline(ach2,100,' ');
string s2(ach2);
//cout << s2 << " ";
ns.s=s2;
int dummy;
istr >> dummy;
//cout << dummy << endl;
ns.length=dummy;
vector<int> vi;
ifs.get(ach,10000,'>');
for(unsigned int i=0;i<10000;++i){
if(!ach[i]) break;
if(ach[i]!='\n')
ns.seq.push_back(ach[i]);
}
vns.push_back(ns);
}
for(unsigned int i=1; i<vns.size();++i){
assert(vns[i].seq.size()==vns[0].seq.size());
}
}
int main(int argc, char* argv[]){
switch (argc){
case 5:{
```

```
ifstream ifs(argv[1]);
if(!ifs){
cerr << "could not open file: " << argv[1] << endl;
exit(1);
}
float start_theta;
sscanf(argv[2], "%f",&start_theta);
//cout << "Theta: (" << start_theta << ',';
float end_theta;
sscanf(argv[3], "%f",&end_theta);
//cout << end_theta << ',';
float delta;
sscanf(argv[4], "%f",&delta);
//cout << delta << ')' << endl;
//exit(0);
vector<named_sequence> v_seq;
read_sequences(v_seq,ifs);
sort(v_seq.begin(),v_seq.end());
vector<vector<vector<char> > > vvvc;
vector<string> vs_locus_names;
vector<int> vi_length;
vector<vector<char> > vvc;
int locus=atoi(v_seq[0].s.c_str());
vs_locus_names.push_back(v_seq[0].s);
vi_length.push_back(v_seq[0].length);
for(unsigned int i=0; i<v_seq.size();++i){
//cout << locus << " " << atoi(v_seq[i].s.c_str()) <<endl;
if(atoi(v_seq[i].s.c_str())==locus)
vvc.push_back(v_seq[i].seq);
else{
```

```
//cout << v_seq[i].s << endl;
locus=atoi(v_seq[i].s.c_str());
vs_locus_names.push_back(v_seq[i].s);
vi_length.push_back(v_seq[i].length);
vvvc.push_back(vvc);
vvc.resize(0);
vvc.push_back(v_seq[i].seq);
}
}
vvvc.push_back(vvc);
//cout << ___LINE___ << endl;
//exit(0);
try{
//BUGBUG: Do somethings with 0 variable sites!
cout << "Theta\tLog_Like" << endl;
for(double theta=start_theta; theta<=end_theta;theta+=delta){
double ll=0.0;
for(unsigned int l=0; l<vvvc.size();++l){
//for(unsigned int i=0; i<vvvc[l].size();++i){
//cout << i << " ";
//for(unsigned int a=0; a<vvvc[l][i].size();++a){
// cout << vvvc[l][i][a];
//}
//cout << endl;
//}
//cout << vs_locus_names[l] << ' ';
//cout << l <<endl;
haplotype_map m_ht;
unsigned int n_haplotypes=vvvc[l].size();
unsigned int n_mutations=vvvc[l][0].size();
```

```
seq2haplotype(vvvc[l],vi_length[l],m_ht);
m_hm old_map_of_htm;
old_map_of_htm[m_ht]=1.0;
CI_mm pmm=old_map_of_htm.begin();
/*
cout << "possible collections of haplotypes at step"
<< 0 << endl;
for(int i=0 ;pmm!=old_map_of_htm.end();++pmm){
cout << "possibility: " << i << endl;
cout << "Prob: " << pmm-> second << endl;
++i;
CI p= pmm->first.begin();
for(;p!=pmm->first.end();++p){
cout << p->second <<' ';
for(unsigned int i=0; i<p->first.size();++i)
cout << p->first[i];
cout << endl;
}
cout << endl;
}
*/
for(unsigned int step=0; step<n_mutations+n_haplotypes-1;++step){
m_hm new_map_of_htm;
CI_mm pmm=old_map_of_htm.begin();
for(;pmm!=old_map_of_htm.end();++pmm){
all_below(pmm->first,pmm->second,vi_length[l]*theta,new_map_of_htm);
//all_below(m_ht,1.0,vi_length[l]*theta,new_map_of_htm);
}
pmm=new_map_of_htm.begin();
/*
```

```
cout << "possible collections of haplotypes at step"
<< (step+1) << endl;
for(int i=0 ;pmm!=new_map_of_htm.end();++pmm){
cout << "possibility: " << i << endl;
cout << "Prob: " << pmm-> second << endl;
++i;
CI p= pmm->first.begin();
for(;p!=pmm->first.end();++p){
cout << p->second <<' ';
for(unsigned int i=0; i<p->first.size();++i)
cout << p->first[i];
cout << endl;
}
cout << endl;
}
*/
old_map_of_htm=new_map_of_htm;
}
//cout << "Locus " << vs_locus_names[l] << " " << vi_length[l]
<< log(old_map_of_htm.begin()->second) << endl;
ll+= log(old_map_of_htm.begin()->second);
}
cout << theta << '\t' << ll << endl;
}
}
catch(tree_error &te){
cerr << "Tree-error: " << te.pchar << endl;
}
catch(...){
cerr << "Something wrong" << endl;
```

```
exit(1);
}
return 0;
}
default: {
cerr << "Usage: tree_forw_step INFILE start_theta end_theta step" << endl;
break;
}
}
}
```

```
######################
FORWARD_TIME.CC
######################

    #include<iostream>
    #include<fstream>
    #include<sstream>
    #include<vector>
    #include<string>
    #include"tree.h"
    #include<algorithm>
    #include<set>
    #include<math.h>
    #include <stdio.h>
    #define MP 22
    #define NP 21 //Maximum value for NDIM=20
    typedef double MAT[MP][NP];
    MAT P;
    double Y[MP],PT[MP];
    int ITER,J,NDIM;
    double FTOL;
    struct named_sequence{
    string s; //name
    vector<char> seq; //sequence
    int length; //length of the sequence
    int operator < (const named_sequence& ns) const {return s<ns.s;}
    //order by name
    };
    void read_sequences(vector<named_sequence>& vns,ifstream& ifs)
    {//sequences must be in fasta format
```

```
vns.resize(0);
for(;;){
char ach[10000];
ifs.getline(ach,10000);
if(!ifs) return;
if(ach[0]!='>'){
cerr << "Not in fasta format!" << endl;
exit(1);
}
named_sequence ns;
string s(&ach[1]);
istringstream istr(s);
char ach2[100];
istr.getline(ach2,100,' ');
string s2(ach2);
//cout << s2 << " ";
ns.s=s2;
int dummy;
istr >> dummy;
//cout << dummy << endl;
ns.length=dummy;
vector<int> vi;
ifs.get(ach,10000,'>');
for(unsigned int i=0;i<10000;++i){
if(!ach[i]) break;
if(ach[i]!='\n')
ns.seq.push_back(ach[i]);
}
vns.push_back(ns);
}
```

```
for(unsigned int i=1; i<vns.size();++i){
assert(vns[i].seq.size()==vns[0].seq.size());
}
}
double log_like(const vector<vector<char> > &vvc,
int length,
double start_theta,
double end_theta,
double time_change)
{
double nu=0.01;
haplotype_map m_ht;
unsigned int n_haplotypes=vvc.size();
unsigned int n_mutations=vvc[0].size();
seq2haplotype(vvc,length,m_ht);
I p=m_ht.begin();
vector<double> vd(1000);
//time_change*=2.0; //compatibity with Hudson's ms
assert(time_change<1000);
vd[0]=1.0;
m_hmvd old_map_of_htm;
old_map_of_htm[m_ht]=vd;
for(unsigned int step=0; step<n_mutations+n_haplotypes-1;++step){
m_hmvd new_map_of_htm;
I_mmvd pmm=old_map_of_htm.begin();
for(;pmm!=old_map_of_htm.end();++pmm){
CI p= pmm->first.begin();
int c_ht=0; //counting haplotypes
for(;p!=pmm->first.end();++p){
c_ht+=p->second;
```

```
}
double factor=1-(c_ht*(c_ht-1)*0.5+length*start_theta*c_ht*0.5)*nu;
unsigned int t=1;
for(; t<time_change;++t){
pmm->second[t]+= pmm->second[t-1]*factor;
}
factor=1-(c_ht*(c_ht-1)*0.5+length*end_theta*c_ht*0.5)*nu;
for(; t<vd.size();++t){
pmm->second[t]+= pmm->second[t-1]*factor;
}
all_below_time(pmm->first,pmm->second,length*start_theta,
length*end_theta,time_change,nu,new_map_of_htm);
}
old_map_of_htm=new_map_of_htm;
}
double sum= 0.0;
for(unsigned int t=0; t<vd.size();++t)
sum+= old_map_of_htm.begin()->second[t];
return log(sum);
}
//user define function
double sum_ll(const vector<vector<vector<char> > > & vvvc,
const vector<int> & vi_length,
double start_theta,
double end_theta,
double time_change)
{
double ll=0.0;
for(unsigned int l=0; l<vvvc.size();++l){
ll+= log_like(vvvc[l], vi_length[l], exp(start_theta), exp(end_theta), exp(time_change));
```

```
}
return ll;
}
void AMOEBA(vector<vector<double> > & P, double *Y, int NDIM, double
FTOL,
int *ITER,const vector<vector<vector<char> > >& vvvc,const vector<int> &vi_length)
{
//const int NMAX=20;
const int ITMAX=10000;
double PR[MP], PRR[MP], PBAR[MP];
double ALPHA=1.0, BETA=0.5, GAMMA=2.0;
int I,IHI,ILO,INHI,J,MPTS;
double RTOL,YPR,YPRR;
MPTS=NDIM+1;
*ITER=0;
e1:ILO=1;
if (Y[1] > Y[2]) {
IHI=1;
INHI=2;
}
else {
IHI=2;
INHI=1;
}
for (I=1; I<=MPTS; I++) {
if (Y[I] < Y[ILO]) ILO=I;
if (Y[I] > Y[IHI]) {
INHI=IHI;
IHI=I;
}
```

```
else if (Y[I] > Y[INHI])
if (I != IHI) INHI=I;
}
//Compute the fractional range from highest to lowest and return if
//satisfactory.
RTOL=2.0*fabs(Y[IHI]-Y[ILO])/(fabs(Y[IHI])+fabs(Y[ILO]));
if (RTOL < FTOL) return; //normal exit
if (*ITER == ITMAX) {
printf(" Amoeba exceeding maximum iterations.\n");
return;
}
*ITER= (*ITER) + 1;
cout<< (*ITER)<<endl;
for (J=1; J<=NDIM; J++) PBAR[J]=0.0;
for (I=1; I<=MPTS; I++)
if (I != IHI)
for (J=1; J<=NDIM; J++)
PBAR[J] += P[I][J];
for (J=1; J<=NDIM; J++) {
PBAR[J] /= 1.0*NDIM;
PR[J]=(1.0+ALPHA)*PBAR[J] - ALPHA*P[IHI][J];
}
YPR=-sum_ll(vvvc,vi_length,PR[1],PR[2],PR[3]);
if (YPR <= Y[ILO]) {
for (J=1; J<=NDIM; J++)
PRR[J]=GAMMA*PR[J] + (1.0-GAMMA)*PBAR[J];
YPRR=-sum_ll(vvvc,vi_length,PRR[1],PRR[2],PRR[3]);
if (YPRR < Y[ILO]) {
for (J=1; J<=NDIM; J++) P[IHI][J]=PRR[J];
Y[IHI]=YPRR;
```

```
}
else {
for (J=1; J<=NDIM; J++) P[IHI][J]=PR[J];
Y[IHI]=YPR;
}
}
else if (YPR >= Y[INHI]) {
if (YPR < Y[IHI]) {
for (J=1; J<=NDIM; J++) P[IHI][J]=PR[J];
Y[IHI]=YPR;
}
for (J=1; J<=NDIM; J++) PRR[J]=BETA*P[IHI][J] + (1.0-BETA)*PBAR[J];
YPRR=-sum_ll(vvvc,vi_length,PRR[1],PRR[2],PRR[3]);
if (YPRR < Y[IHI]) {
for (J=1; J<=NDIM; J++) P[IHI][J]=PRR[J];
Y[IHI]=YPRR;
}
else
for (I=1; I<=MPTS; I++)
if (I != ILO) {
for (J=1; J<=NDIM; J++) {
PR[J]=0.5*(P[I][J] + P[ILO][J]);
P[I][J]=PR[J];
}
Y[I]=-sum_ll(vvvc,vi_length,PR[1],PR[2],PR[3]);
}
}
else {
for (J=1; J<=NDIM; J++) P[IHI][J]=PR[J];
Y[IHI]=YPR;
```

```
}
goto e1;
}
int main(int argc, char* argv[]){
NDIM=3; // 3 variables
FTOL=1e-8; // User given tolerance
//define NDIM+1 initial vertices (one by row)
vector<vector<double> > P(NDIM + 2);
P[1].resize(4);
P[1][1]= log(0.0002); P[1][2]=log(0.002); P[1][3]=log(100);
P[2].resize(4);
P[2][1]= log(0.002); P[2][2]=log(0.0002); P[2][3]=log(100);
P[3].resize(4);
P[3][1]= log(0.005); P[3][2]=log(0.0003); P[3][3]=log(100);
P[4].resize(4);
P[4][1]= log(0.0003); P[4][2]=log(0.005); P[4][3]=log(50);
switch (argc){
case 2:{
ifstream ifs(argv[1]);
if(!ifs){
cerr << "could not open file: " << argv[1] << endl;
exit(1);
}
vector<named_sequence> v_seq;
read_sequences(v_seq,ifs);
sort(v_seq.begin(),v_seq.end());
vector<vector<vector<char> > > vvvc;
vector<string> vs_locus_names;
vector<int> vi_length;
vector<vector<char> > vvc;
```

```
int locus=atoi(v_seq[0].s.c_str());
vs_locus_names.push_back(v_seq[0].s);
vi_length.push_back(v_seq[0].length);
for(unsigned int i=0; i<v_seq.size();++i){
if(atoi(v_seq[i].s.c_str())==locus)
vvc.push_back(v_seq[i].seq);
else{
locus=atoi(v_seq[i].s.c_str());
vs_locus_names.push_back(v_seq[i].s);
vi_length.push_back(v_seq[i].length);
vvvc.push_back(vvc);
vvc.resize(0);
vvc.push_back(v_seq[i].seq);
}
}
vvvc.push_back(vvc);
try{
for (int I=1; I<=NDIM+1; I++) {
PT[1]=P[I][1];
PT[2]=P[I][2];
PT[3]=P[I][3];
Y[I]=-sum_ll(vvvc,vi_length,PT[1],PT[2],PT[3]);
cout<< PT[1] << " " << PT[2] << " " << PT[3] << " " << Y[I] << endl;
}
AMOEBA(P,Y,NDIM,FTOL,&ITER,vvvc,vi_length);
printf(" %d", ITER);
// printf(" Best NDIM+1 points:\n");
//for (int I=1 ; I<=NDIM+1; I++) {
// printf("%d", ITER);
for (int J=1; J<=NDIM; J++)
```

```
printf(" %f", exp(P[1][J]));
printf(" %14.10f", Y[1]);
// printf("\n");
//}
//printf("\n Best NDIM+1 Maximum values:\n");
//for (int I=1; I<=NDIM+1; I++)
//printf(" %14.10f", Y[1]);
//printf("\n");
}
catch(tree_error &te){
cerr << "Tree-error: " << te.pchar << endl;
}
catch(...){
cerr << "Something wrong" << endl;
exit(1);
}
return 0;
}
default: {
cerr << "Usage: tree_forw_update infile.fst" << endl;
break;
}
}
}
```

Appendix III: Abstract

In den letzten zehn Jahren wurden bemerkenswerte Fortschritte in der Biologie gemacht. Biologen, die natürliche Populationen von Pflanzen oder Tieren studieren, haben Zugriff auf neue Technologien wie das Next Generation Sequencing. Häufig müssen rechenintensive statistische Verfahren für die Analyse von komplexen biologischen Daten entwickelt werden. Der Fortschritt in der Computertechnik erlaubt es, rechenintensive statistische Analysen auf Desktop-Computern auszuführen. Dies führte zu einem signifikanten Fortschritt bei der Entwicklung statistischer Verfahren in der Genetik, wie etwa die Verwendung von Monte Carlo und Markov Chain Monte Carlo (MCMC) Methoden zur Berechnung der Likelihoods und a-posteriori Wahrscheinlichkeiten. Diese Dissertation konzentriert sich auf die Entwicklung von Likelihood-Methoden und Likelihood-freier statistischen Verfahren und deren Anwendung auf die Analyse von genetischen Daten. Zunächst schlagen wir eine effiziente Methode zur Berechnung der Likelihood vor, die für die Schätzung von zeitabhängigen Mutationsraten im Infinite-Sites Mutationsmodell verwendet werden können. Im Rahmen der Likelihood-freien Methoden schlagen wir eine Methode zur Auswahl von Statistiken vor, die dann im Rahmen des „Approximate Bayesian Computation“ (ABC)-Algorithmus verwendet werden. Ziel ist es, die summary Statistiken so zu wählen, dass die tatsächliche a posteriori Verteilung möglichst gut approximiert wird. Das vorgeschlagene Verfahren, basierend auf der „least angle regression“ (LAR), ist besser in Bezug auf Rechenzeit und Genauigkeit als vergleichbare Methoden in der Literatur. Wir schlagen auch Methoden, um den Akzeptanz-Cutoff für ABC zu bestimmen vor und vergleichen diese.

Appendix IV: Curriculum Vitae

NAME	**Muhammad FAISAL**
ADDRESS	Brünner Straße 72/3/236, 1210 Vienna (Austria)
DATE OF BIRTH	09 September 1982
EDUCATION	M.Sc. (Statistics) 2004 – 2006
	Bahauddin Zakariya University, Multan (Pakistan)
	Thesis Title: *Reduction of Sample Selection Bias under Multicolinearity*.
	B.Sc. (Statistics, Mathematics, Advance Computer Studies) 2002 – 2004
	ICS (Statistics, Mathematics, Computer Science) 2000-2002
	S.S.C 1998 – 2000
CONFERNCE PUBLICATIONS	Akbar, A. Pasha, G. R., Faisal, M. and Aslam, M. (2007). Reduction in Sample Selection Bias Under Multicolinearity. Proceedings of 3rd National Conference.Vol. 14, 187-194. ISBN: 978-969-8858-02-5. ISlamic Countries Society of Statistical Sciences.
JOURNAL PUBLICATIONS	Khan M. S., Awan M. S., Faisal M., Nadeem F., Leitgeb E., and Mathiopoulos T. M. (2010). Probabilistic Model for Free-Space Optical Links Under Continental Fog Conditions. Radioengineering. Vol. 19, No. 3.
	Hussain I., Spoeck G., Pilz J., Faisal M., and Yu H. (2011) . Hierarchical Bayesian Spatio-Temporal Interpolation including Covariates: during Monsoon Periods in Pakistan. Pakistan Journal of Statistics (submitted)
	Iqbal F., Hoeger H., Voigtlaender T., Lubec G., Zehetmayer S., Isbrandt D., Muehl A., Faisal M., Bodamer O. A., (2011). Guanidinoacetate N-Methyltransferase deficient mouse as a model for neuroprotective role of Creatine following neonatal hypoxic-ischemic encephalopathy. Behavioural Brain Research (submitted)
	Faisal M., Futschik A. (2011). Choosing Summary Statistics for ABC (submitted)
	Faisal, M., Claus, V., and Futschik, A. (2011). Exact Likelihood Computaion of Mutations rates under Infinite Sites Model.(submitted)
	Khan, M. S., Muhammad, S. S., Awan, M. S., Kvicera, V., Grabner,M., Leitgeb, E., and Faisal, M. (2011). Further Results on Fog Modeling for Terrestrial FSO Links. Jounral of Optical Engneering. (submitted)