RICE UNIVERSITY

Stochastic Models and Linkage Disequilibrium: Estimating the Recombination Coefficient

by

Vernon Shane Pankratz

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

Doctor of Philosophy

Approved, Thesis Committee:

Marek Kimmel, Professor, Chair
Statistics

Ranajit Chakraborty, Allen King Professor
Center for Human Genetics, UT-Houston

Keith A. Baggerly, Assistant Professor
Statistics

Kathleen S. Matthews, Stewart Professor
Biochemistry and Cell Biology

Houston, Texas

June, 1998

Abstract

Stochastic Models and Linkage Disequilibrium: Estimating the Recombination Coefficient

by

Vernon Shane Pankratz

By studying the rate of recombination between genetic markers and disease genes with linkage analysis, scientists have successfully mapped the locations of disease-influencing genes to within one centiMorgan. However, one centiMorgan corresponds to a sequence of about one million base pairs of DNA, which is prohibitively large for a physical search for a specific gene. Therefore, other genetic mapping techniques are needed to define search regions that are small enough for physical mapping techniques to be feasible. One such method is called linkage disequilibrium mapping. Linkage disequilibrium can serve as a complement, or even an alternative, to linkage analysis. It is capable of estimating genetic distances that are as small as tens of kilobases of DNA, a great improvement over the resolution of linkage analysis. However, one must describe the joint transmission of disease genes and linked marker loci through many generations in order to use linkage disequilibrium for genetic mapping purposes. This thesis examines two classes of population models, Galton-Watson branching processes and Moran/Coalescent models, within the framework of linkage disequilibrium. That is, it uses moments of allele frequencies derived from these models to form approximate likelihood functions for the recombination rate. These likelihoods make it possible to estimate the location of a disease-influencing mutation, particularly when the likelihoods from several markers within a small region of DNA are combined to form a composite likelihood. Application of this composite likelihood methodology to both simulated and published data demonstrates that linkage disequilibrium mapping can be successfully used for fine-scale mapping purposes.

Acknowledgments

This thesis would not be have been possible were it not for my wife, Stephanie, whose support and good advice saved the day many times.

I am also indebted to my advisor, Marek Kimmel, for all of his advice and for help in crucial points throughout this thesis.

I also need to thank my committee for their patience with me, as well as for their encouragement.

This work was funded by the **Keck Center for Computational Biology** through the U.S. National Library of Medicine training grant NLM 1T15LM07093-04.

Contents

Illustrations

Tables

Preface

In the nineteenth century, Gregor Mendel performed his famous experiments in which he formed crosses of several varieties of peas and observed the resulting offspring. The experimental results led him to propose that genes govern heritable characteristics and that these genes are transmitted from parents to offspring in a predictable way.

We now know that one of Mendel's hypotheses was not entirely correct. He postulated that the transmission of alleles from different genes is independent. We now know that the alleles of genes, or loci, that are close together on the same chromosome are often passed on in tandem. In fact, the alleles of different loci on the same chromosome are transmitted as a unit unless the two homologous strands exchange DNA through a process called recombination.

While we do not fully understand the mechanisms that govern recombination, we know that the average number of recombination events occurring in an interval is inversely proportional to the physical length of that interval. This preliminary knowledge has proven to be very useful in establishing genetic maps of the chromosomes of various organisms. Since these genetic maps compare favorably with physical maps, it is possible study the process of recombination to search for genes that govern specific traits.

This search is traditionally executed using a technique called linkage analysis. The concept of linkage analysis is simple; one collects genetic pedigrees and "counts" the number of recombinations that occur between the disease gene and various marker loci. This method has been successful in mapping genes that influence a variety of diseases to regions as small as one million base pairs (1 Mb) of DNA.

Researchers would like to be able to map genes to regions smaller than 1Mb, however in practice it is very difficult to do so using linkage analysis. The limitation arises from the estimation procedure used in linkage analysis: one "counts" recombination events within extended families. In order to estimate recombination rates of less than one recombination in one hundred meioses, one must examine at least several hundred meioses. Hence, obtaining samples of sufficient size to perform fine-scale mapping with linkage analysis is very difficult, and often impossible.

Many researchers are looking to population-based methods in a search to obtain finer resolution in genetic maps. These population-based methods depend on linkage disequilibrium, a term from population genetics that refers to the situation that exists when two loci do not pass on their alleles independently. The difficulty that arises with linkage disequilibrium is that its behavior depends on the properties of the population under consideration.

The intent of this thesis is to use models of population genetics to obtain information about linkage disequilibrium as it relates to the distance between linked loci. The models provide this information when they are adapted to describe the joint behavior of loci in real populations. Moments from these models make it possible to obtain approximations to likelihoods which can estimate genetic distances as small as tens of thousands of base pairs. These distances are orders of magnitude less than the minimal distances that linkage analysis can provide.

Chapter 1

Introduction

The purpose of this thesis is to study the use of linkage disequilibrium for the fine-scale mapping of disease genes. Modeling the distribution of marker alleles as they evolve within the disease population is of particular interest, as this makes it possible to obtain maximum-likelihood estimates of the recombination coefficient.

Chapter 2 begins the study by introducing the topic of genetic mapping. It first discusses some basic genetic concepts by reviewing issues dealing with Mendelian inheritance and recombination. It then introduces the topics of linkage analysis and linkage disequilibrium. Its concluding section introduces several models that describe the perpetuation of genetic populations.

The third chapter goes into more detail about the use of linkage disequilibrium in the area of fine-scale mapping. It discusses various measures of linkage disequilibrium before moving on to mention current methods for genetic mapping with linkage disequilibrium. The most important of these are maximum-likelihood based procedures that estimate the position of disease genes within maps consisting of one or more marker loci.

After a description of Galton-Watson branching processes and the Moran/Coalescent model in the first part of the fourth chapter, the thesis contains original work. It discusses progress that can be made by modeling the expansion of the disease population within a larger population.

The remainder of Chapter 4 demonstrates how to obtain the moments from each of the two classes of models. It also presents a general framework for modeling the joint behavior of a disease gene linked to a marker locus, with or without mutations at the disease and marker loci.

Chapter 5 applies the methods in Chapter 4. It presents methods for forming approximate likelihoods that estimate the recombination rate between the disease gene and a single marker locus.

Chapter 6 provides information about linkage disequilibrium mapping with respect to more than one genetic marker. It makes some headway for the case where there are two markers, but primarily relies on composite likelihood methodology to combine single-marker likelihoods.

Chapter 7 contains evaluations of the methodology developed in previous chapters. Simulation results indicate that the methods can be used to successfully map disease genes. Application of the methods to published data from Cystic Fibrosis and Huntington's Disease verify the simulation results. The estimated locations of the disease genes were less than 100 kb away from known mutations within the disease genes. If fact, the estimate for the location of the Huntington's Disease gene was only 5 kb from the truth.

Chapter 8 summarizes the work of this thesis and presents several possible areas for future work.

Chapter 2

Genetic Mapping

This chapter introduces the use of genetic data to locate genes within a genome. First, it describes basic principles of Mendelian inheritance. Second, it addresses issues arising when two or more genes are linked. It finally outlines the two major tools used to map human genetic diseases: linkage analysis and linkage disequilibrium.

2.1 Mendelian Inheritance

Gregor Mendel studied the transmission of features from parents to offspring by crossing various varieties of peas that were pure for specific traits. The results from his experiments led him to propose that an organism has distinct hereditary units that govern specific characteristics and that those units do not blend within the parents, but rather are discrete items that segregate during the formation of sex cells, or *gametes*.

Many multicellular organisms such as peas and humans are *diploid*, that is, they have two sets of genes. The specific forms of the gene, known as *alleles*, govern the expression of different traits. The pairs of alleles for a single gene, or a collection of genes, constitute the *genotype*. The genotype for a single gene is *homozygous* if the diploid pair consists of two identical alleles and *heterozygous* if it has two different alleles. The *phenotype* is the physical trait expressed by the genotype. A phenotype is *recessive* if expression of the trait occurs only when the gene is *homozygous* for the allele. A trait is *dominant* if it can be expressed when only one copy of the allele is present.

Genes are not scattered haphazardly throughout the cell. Rather, they reside within the nucleus within organelles called *chromosomes.* The collection of all chromosomes within a single cell of an organism constitutes the organism's *genome.* It is assumed that the genome of multi-cellular organisms does not vary from cell to cell.

The genome of diploid organisms consists of two complete sets of chromosomes, one receivedfrom each of their parents. The elements of a pair of chromosomes are called *homologous* since they are similar but not identical. Genes, in principle, are arrayed in linear sequences on chromosomes, occupying approximately the same position for individuals from the same species. The word *locus* refers to the position of a gene, or any sequence of DNA, in the genome. A *haplotype* is the ordered vector of alleles at a collection of loci on one homolog, and represents the genetic information contributed by one parent.

Homologs segregate to separate sex cells in *meiosis.* Prior to meiosis, the chromosomes replicate so that each homolog consists of two identical strands of DNA, called *chromatids.* This bundle of four strands of DNA maintains a primary point of contact at the *centromere.* The chromatids become entangled at other locations as well. As the chromatids separate, some of these points of contact are conserved and form what are known as *chiasmata* (see *e.g.* [1]). At these chiasmata, homologous chromatids exchange genetic material. The chromatids experiencing such a *crossover* or *recombination* event no longer consist of the original sequence of DNA, but of combined fragments of the two ancestral strands, called *recombinant* DNA (see *e.g.* [58]). Figure 2.1 shows a schematic representation of the events leading to the placement of chromosomes into gametes. First, the two homologous

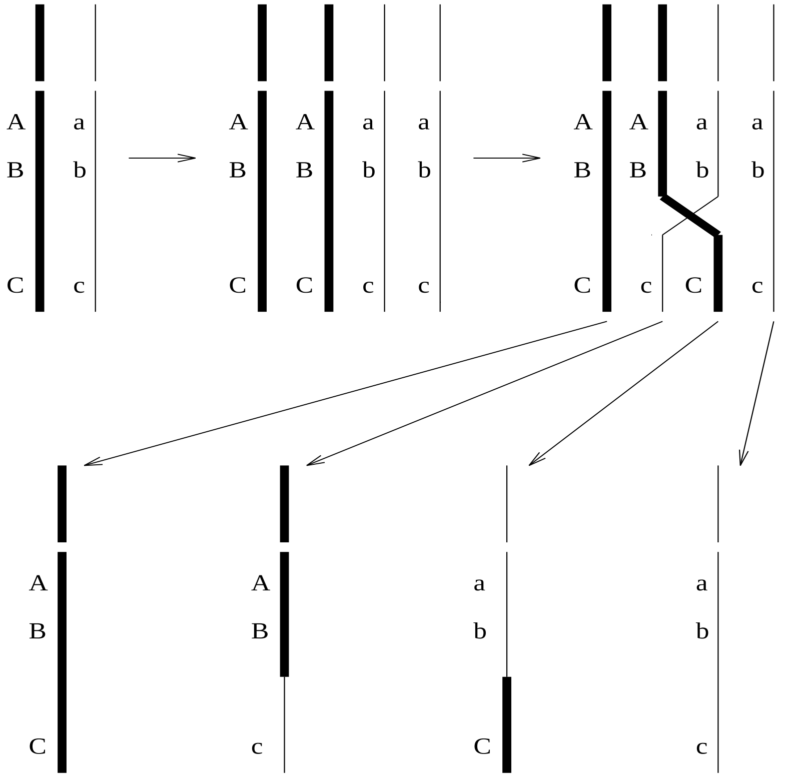

Figure 2.1 Schematic representation of a single recombination event between two homologous chromosomes carrying three genes.

strands replicate. As they separate, one or more chiasmata form. As the haploid strands of DNA segregate into separate sex cells, some of the chiasmata force an exchange of DNA between the strands to produce recombinant, or *recombined* DNA.

2.2 Recombinatorial Distance

The study of recombination events has led to a technique known as genetic mapping, which positions loci on chromosomes based on the frequency of crossovers that occur between them. This technique is possible because the physical distance between two loci is a good predictor of the rate of genetic recombination. The main idea is that crossovers between two loci occur at a rate that is inversely proportional to the length of DNA separating them. Therefore, if two loci are close together, we will rarely observe a recombination event between them. One way to justify this is to consider the rate of crossovers in human meioses. For example, the human genome consists of approximately 3×10^9 base pairs of DNA and the average number of recombination events per meiosis is about 33 [61]. Therefore, if recombination events occur at random throughout the genome, then on average, 10^8 base pairs separate each recombination event.

2.2.1 Map Distances

When recombination events between two loci on the same chromosome occur at a rate of 50 percent, the segregation pattern of the two loci is identical to that expected when two loci occupy positions on separate chromosomes. This is so because only one-half of the recombined genes are observable within families when recombinations occur freely. If recombinations occur between two loci in fewer than half of all meioses, then the loci are said to be *linked*. The degree of linkage is parameterized by the *recombination fraction*, or the percentage of meioses with recombinations between two loci. Henceforth, we will use r to represent this parameter, although we note that θ is also frequently used (see it e.g. [58]). Since r is a parameter for the degree of linkage, its support is the interval [0, 0.5].

The recombination rate between two loci can serve as a stochastic measure of distance between two loci. The genetic *map distance* between two loci is defined as the expected number of crossovers occurring between them on a single chromatid [58]. This expectation is scaled in units of Morgans. One Morgan measures a length of DNA that, on average, experiences one crossover per meiotic event. The recombination fraction can be thought of as the probability of experiencing a recombination event in a sequence of DNA, with one-half lost due to segregation. Therefore, it is possible to translate the recombination fraction into units of Morgans. This can be done in a variety of ways, as we discuss in the next section. For the simplest case we observe that one centiMorgan (cM), is approximately equivalent to a recombination fraction of one percent, as it measures a length of DNA that has an average of one recombination event per one hundred meioses. Once we have verified both the linear order of loci on a chromosome and the distances between the loci, we have defined a *genetic map*.

We can use a genetic map to approximate a physical map. There are on average 33 crossovers per human meiosis. Hence the human genome is approximately 33 Morgans in length. The physical length of the human genome is approximately three billion base pairs. Equating these two quantities, we discover that one centi-Morgan, or a recombination fraction of $r = 0.01$, corresponds to about one million base pairs.

2.2.2 Mapping Functions

Genetic maps do not correspond directly to physical maps. At least two factors influence the discrepancy: multiple crossovers and interference. The first factor influences genetic map distances because multiple crossovers in an interval may be indistinguishable from fewer recombination events. The second factor comes into play if recombination events do not occur independently throughout the genome.

An important parameter in the description of interference is the *coefficient of coincidence*, c, which is defined as the ratio of observed to expected double recombination events. In order to obtain a parameterization of c, consider three loci in the order A-B-C, with the pairwise recombination rates r_{AB}, r_{BC} and r_{AC}. We now examine the recombination in the intervals A-B and B-C jointly.

With respect to the joint recombination in A-B and B-C, we can observe four classes of offspring: double recombinants with recombination in A-B and B-C, single recombinants with recombination in A-B but not B-C, single recombinants with recombination in B-C but not A-B and nonrecombinants. If we denote the probabilities of these events occurring as g_{11}, g_{10}, g_{01} and g_{00} respectively, then we can estimate the pairwise recombination events as

$$\begin{aligned} r_{\text{AB}} &= g_{11} + g_{10}, \\ r_{\text{BC}} &= g_{11} + g_{01}, \\ r_{\text{AC}} &= g_{10} + g_{01}. \end{aligned} \tag{2.1}$$

Using these equations, we can obtain an estimate of the probability of observing a double recombinant:

$$g_{11} = \frac{r_{\text{AB}} + r_{\text{BC}} - r_{\text{AC}}}{2}. \tag{2.2}$$

This leads to the parameterization of c proposed by Muller [56] in 1916,

$$c = \frac{r_{\text{AB}} + r_{\text{BC}} - r_{\text{AC}}}{2\ r_{\text{AB}}\ r_{\text{BC}}}, \tag{2.3}$$

since the probability of observing a double recombinant under independence is $r_{\text{AB}}\ r_{\text{BC}}$. The possible range of c can be shown to be

$$\max\left(0, \frac{r_{\text{AB}} + r_{\text{BC}} - 1/2}{2 r_{\text{AB}} r_{\text{BC}}}\right) \leq c \leq \min\left(\frac{1}{r_{\text{AB}}}, \frac{1}{r_{\text{BC}}}\right), \tag{2.4}$$

(see it e.g. [58]).

Interference, or the extent to which a recombination events influence the occurrence or nearby recombination events, is related to the concept of coincidence. It is mathematically defined as $I = 1 - c$ [1]. From this relationship, we can see that interference is absent when $c = 1$, and can be either positive or negative depending on the amount of coincidence. In most species, interference seems to be positive, so that one usually assumes $c \leq 1$, with $c = 0$ implying complete interference [58].

Mapping functions make it possible to obtain better information concerning genetic distance by accounting for the factors mentioned above. Three mapping functions are commonly used for human genetic maps. They are the Morgan, Haldane and Kosambi mapping functions.

The Morgan map function [54] is the simplest of the three. If m is the map distance and r is the recombination coefficient, we can write the Morgan map function as

$$m = \begin{cases} r, & 0 \leq r < \frac{1}{2}, \\ \infty, & \text{otherwise.} \end{cases} \tag{2.5}$$

This equation is appropriate when multiple crossovers between two loci do not occur. Therefore, it is appropriate to use this mapping function for tightly linked loci.

Multiple crossovers can occur between two distant loci, however. In 1919, Haldane [18] assumed that crossovers in different intervals occur according to a

Poisson point process. This assumption leads to the Haldane map function,

$$m = \begin{cases} -\frac{1}{2}\ln(1-2r), & 0 \le r < \frac{1}{2}, \\ \infty, & \text{otherwise}, \end{cases} \tag{2.6}$$

whose inverse is

$$h(r) = \frac{1}{2}\left(1 - \exp(-2m)\right). \tag{2.7}$$

This mapping function is used extensively later in this thesis, in keeping with conventions set by others (see it e.g. [58]).

The Morgan and Haldane map functions represent two extremes. The Morgan map function essentially assumes complete interference and the Haldane map function assumes interference to be absent. It is possible to derive different mapping functions by assuming different levels of interference. In 1944, Kosambi [45] built a mapping function built on the premise that one recombination event must occur on a chromosome, and that no interference exist among subsequent recombination events. This assumption led to what we now call the Kosambi map function:

$$m = \frac{1}{2}\tanh^{-1}(2\,r). \tag{2.8}$$

Each of these map functions conform to a differential equation first obtained by Haldane [18]:

$$\frac{\mathrm{d}r}{\mathrm{d}m} = 1 - 2\,c\,r, \tag{2.9}$$

where c is defined in Equation 2.3. The distinct map functions arise by changing the form of c in the differential equation. For example, if $c = 0$ we obtain the

Morgan mapping function and if $c = 1$ we obtain the Haldane mapping function. Likewise, if we set

$$c \;=\; 2\,\frac{r_{\mathrm{AB}} + r_{\mathrm{BC}}}{1 + 4r_{\mathrm{AB}}\,r_{\mathrm{BC}}}, \tag{2.10}$$

we obtain the Kosambi mapping function. Other researchers, such as Carter and Falconer [6], have obtained different map functions by modeling c in various ways. Rao *et al.* [60] made a valuable contribution by incorporating existing map functions into a family of models, making it possible to estimate the map function from data.

2.3 Linkage Analysis

The purpose of linkage analysis is to estimate the recombinatorial distance between two loci, often a marker locus and an unknown disease gene. In its simplest form, linkage analysis estimates the recombination coefficient by dividing the total number of observed recombinations by the total number of possible recombinations in a pedigree or set of pedigrees. However, recombined DNA is not always recognizable, so more sophisticated estimation techniques are needed [69].

Researchers use maximum-likelihood methods to overcome the problems that present themselves in linkage analysis. The likelihood function contains information concerning the probability of a phenotype given its genotype (*penetrance*), as well as information concerning the probability of the genotype given the recombination coefficient and the genotypes of all ancestors (see *e.g.* [58]). Once the mode of inheritance, penetrance, allele frequencies and other genetic information have been specified, it is possible to explicitly write a formula for the probability of a disease/marker haplotype that depends on the recombination coefficient. The formulae thus obtained are then aggregated for all the data to form a likelihood equation that can be used to obtain an estimate of r.

Linkage analysis has been used with great success to obtain coarse genetic maps. The granularity of these maps is on the order of centiMorgans, or millions of base pairs (see it e.g. [58]). As noted earlier, a genetic distance of 1 cM corresponds to a recombination fraction of approximately 0.01. This recombination rate is essentially a lower limit on the resolution of linkage analysis. This limitation arises due to the fact that linkage analysis estimates r by counting, or inferring, recombination events. It is very difficult to collect enough family-based data to reliably observe recombinations that occur with a frequency lower than one out of one hundred (see it e.g. [11]). For instance, if $r = 0.001$ and we wish to observe at least one recombined haplotype with 90% probability, we must observe more than 2300 meioses where recombination events are identifiable.

2.4 Linkage Disequilibrium

Since linkage analysis suffers from resolution constraints imposed by sample size considerations, it is necessary to use other techniques for fine-scale mapping. One of the alternatives is linkage disequilibrium. Linkage disequilibrium, or allelic association, refers to the situation where different loci do not pass on their alleles independently. Many genetic factors can influence this. Some of these factors are physical distance between loci, population admixture, nonrandom mating and mutation (see it e.g. [22]). Since physical proximity has an effect on linkage disequilibrium, it can provide information about the recombination coefficient. However, one must be cautious when using linkage disequilibrium to form genetic maps, since linkage disequilibrium can be caused, and maintained, by more than just linkage.

To illustrate the concept of linkage disequilibrium, consider two loci: locus A with k_a alleles and locus B with k_b alleles. Assume that allele i of locus A occurs with frequency p_i, and that allele j of locus B has a population frequency of q_j. Also, let P_{ij} denote the joint frequency of allele i of locus A and allele j of locus B. The two loci are in a state of linkage disequilibrium if the joint allele frequencies do not equal the product of the marginal allele frequencies, i.e. if $P_{ij} \neq p_i\, q_j$ for some $i \in \{1, \ldots, k_a\}, j \in \{1, \ldots, k_b\}$. Hence, we can test for linkage disequilibrium using the methologies of classical contingency table analysis.

We often wish to measure the magnitude of linkage disequilibrium, or the deviation from independent transmission of alleles. In the situation where there are two loci, each with two alleles, the amount of disequilibrium is commonly measured by some function of

$$D = P_{11}\, P_{22}\; -\; P_{21}\, P_{22}. \tag{2.11}$$

Note that D is a function of the odds ratio, a measure that is often used in categorical data analysis (see it e.g. [3]).

One of the appealing features of linkage disequilibrium is that recombination causes it to behave in a predictable manner as populations evolve from generation to generation. In fact, recombination events decay disequilibrium by a factor of $1 - r$ per generation.

We can establish this result for the measure defined in Equation 2.11 if we make additional assumptions. If we assume that individuals mate at random and that there is no mutation at either locus, we find that the joint frequencies after one generation are

$$P_{ij}(t+1) = P_{ij}(t) - r[P_{ij}(t) - p_i q_j]. \tag{2.12}$$

Using this identity, we obtain the long-known result [35] [63] that after t generations,

$$D_t = (1 - r)^t D_0. \tag{2.13}$$

Thus we see that allelic associations are greatly influenced by the recombinatorial distance between loci. Although recombination diminishes the degree of linkage disequilibrium, many population effects can also play a significant role in the existence of linkage disequilibrium (see it e.g. [22]). For this reason, many researchers are hesitant to utilize it to map disease genes.

However, if we are careful in our application of linkage disequilibrium, it can be a powerful tool for mapping disease genes. It can be especially useful in refining the location of disease genes whose position has been roughly identified through linkage analysis. In fact, linkage disequilibrium mapping has successfully localized genes for a variety of diseases in spite of its shortcomings (see it e.g. [11]).

2.5 Population Models

The behavior of linkage disequilibrium within a population depends on the manner in which the population evolves. Therefore, if we are to successfully utilize linkage disequilibrium as a tool for mapping disease genes, we must consider the evolution of populations. This can be a very difficult problem. However, we can turn to the field of population genetics, where many models of population behavior exist that capture various features of human population dynamics. In this section, we will discuss several of these.

2.5.1 Wright-Fisher Model

The Wright-Fisher (or Fisher-Wright) model was studied concurrently by Wright [75] and Fisher [16] in the first part of the 20th century. Figure 2.2 is an illustration of the behavior of the model as it evolves through time. This model requires three primary assumptions. First, it assumes that time is counted by generations. Second, it assumes that the population of chromosomes maintains a fixed size

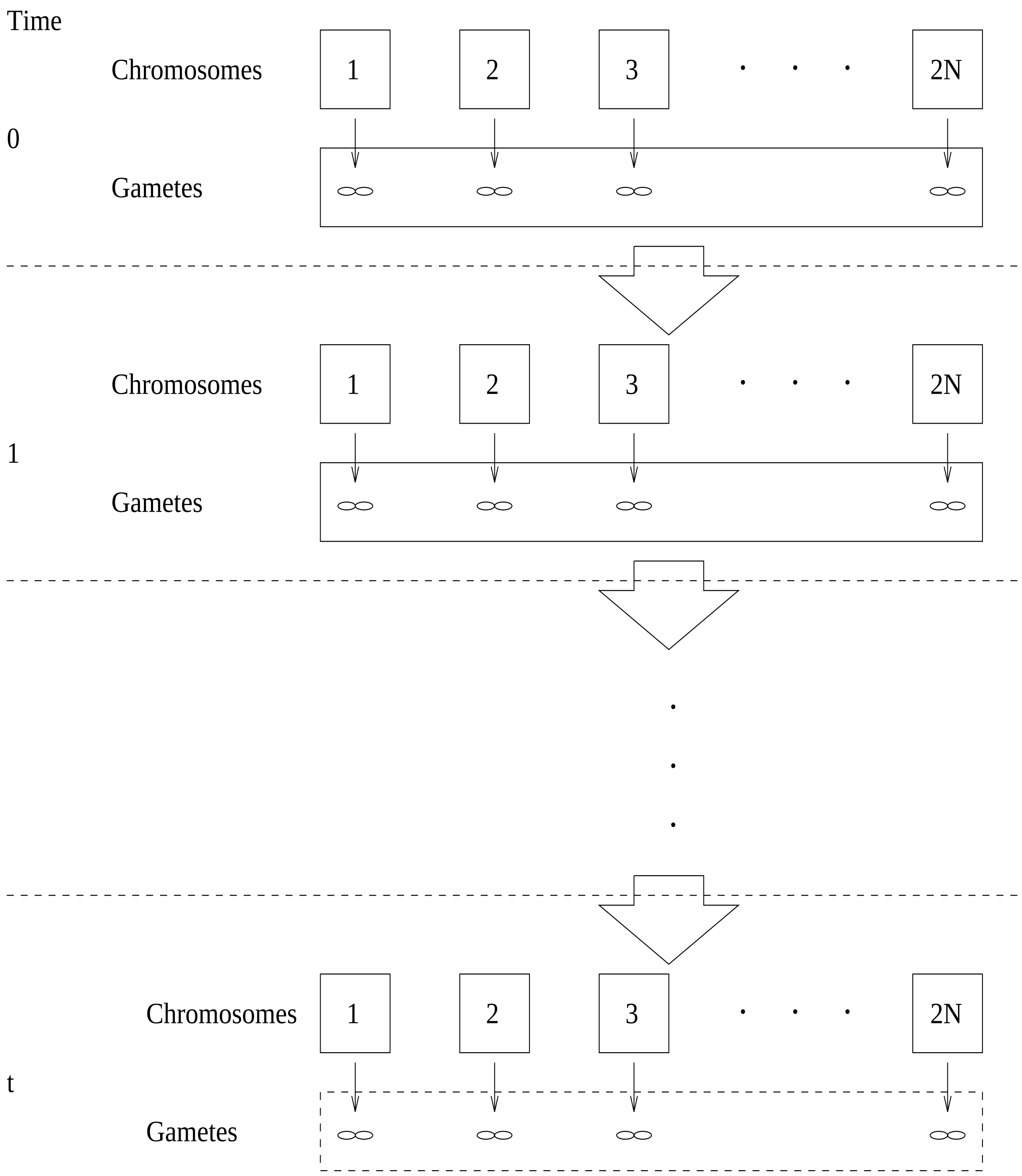

Figure 2.2 Illustration of the Wright-Fisher model of evolution for a population with N diploid individuals.

through time. The third feature deals with the production of offspring from one generation to form the next. In the reproductive step, each chromosome produces a large number of gametes. The chromosomes in the next generation are drawn at random from the total pool of gametes. This mechanism is equivalent to a multinomial sampling scheme, where chromosomes in one generation are sampled with replacement from those in the previous generation.

We now point out two issues that arise in considering this model. First, each chromosome is treated identically. Hence, when applying the Wright-Fisher model to a diploid population with N individuals, one must consider $2N$ chromosomes, with no reference to the diploid nature of the individuals in the population. This is only an approximation to the actual biological process of reproduction, but if the population is large enough this approximation works well.

The second feature is the *fixation* of chromosomes, or alleles. Because not all chromosomes in the population in one generation produce offspring in the generation, due to multinomial sampling, eventually a single chromosome will be fixed in the population. This implies that one can look backward in time and trace the genealogy of the current population to a single ancestor chromosome. This idea led to the proposal of an object called the *coalescent*. It was first suggested by Ewens [13] and developed by Kingman [41] [42] and Tajima [67].

To understand the reasoning behind the idea of the coalescent, consider two chromosomes randomly sampled from a population of size $2N$. The probability that they both share a common ancestor in any preceding generation is $1/2N$ due to the multinomial sampling mechanism. Repeating this argument for more than one generation, we can find the probability of coalescence in t generations. This probability is $(1/2N)[1-(1/2N)]^t$, or the probability that one common ancestor existed t generations in the past multiplied by the probability that there were no common ancestors more recent than t. Note that this probability can be approximated by the exponential distribution with mean $2N$ since $[1-(1/2N)]^t \approx \exp(t/2N)$.

2.5.2 Time-Continuous Moran Model

As an alternative to the Wright-Fisher model, we now mention a generalization of a discrete-time model proposed by Moran [53]. The original Moran model describes the evolution of $2N$ haploid individuals. It differs from the Wright-Fisher model in its description of the reproduction process. Rather than assuming that the population reproduces in non-overlapping generations, it assumes that population changes occur at fixed time points $t = 0, \Delta t, 2\Delta t, \ldots$. At these points, one new element is added to the population and one element is lost. The genetic makeup of the new member is obtained by randomly sampling from the already-existing population. The individual who leaves the population is chosen at random from those present immediately prior to the most recent birth. Therefore the probability that any of these elements dies is equal to $1/2N$. This has the effect that the lifetimes of the particles in the population follow a geometric distribution with a mean of $2N$ time units.

One can extend the Moran model to operate in continuous time by allowing a random length of time to pass between the times of birth/death and by using the same birth/death scheme as in the time-discrete version. The most direct translation from to discrete to continuous time is to let time between deaths follow the exponential distribution, the time-continuous analog to the geometric distribution. This allows us to describe the time points with a Poisson point process, and facilitates the study of the time-continuous version.

The Moran model has some features that make its use appealing. First, as we have previously noted, we can construct the model so that it operates in continuous time. Second, since we can specify the population at a randomly chosen time based on the makeup of the population at the most recent known point in the past, the Moran model is a Markov process. Third, since a single birth occurs at each time point, one can study the model by running time in reverse [13]. This allows us to derive results from this model using the coalescent.

2.5.3 Galton-Watson Branching Process

Another class of models can be applied within the framework of population genetics. These models are purely stochastic and are referred to as branching processes. A Galton-Watson process is the simplest type of branching process, which we can describe as follows. A population begins at generation 0 with a single particle. At the commencement of the first generation, this original particle splits into a number of identical offspring. The number of offspring is governed by a probability density on the non-negative integers. Each of the progeny then independently splits into a number of offspring at the inception of the next generation. The number of offspring produced by each particle is determined by the same probability density as the one corresponding to the founder of the population (see it e.g. [62] or [21]).

The properties of the offspring distribution govern the behavior of the branching process. The simplest characterization is the average behavior of the process. Branching processes are called *subcritical*, *critical* or *supercritical* if the means of their offspring distributions are less than, equal to, or greater than one, respectively. Subcritical and critical processes become extinct with probability one. Supercritical processes exhibit exponential growth on average, and have a non-zero probability of extinction.

One model useful for linkage disequilibrium mapping is due to Kaplan *et al.* [36]. They used a branching process model with Poisson offspring distributions to model the propagation of disease genes within a larger population. This model can be viewed as an approximation to the propagation of a rare disease gene through genetic drift within the Wright-Fisher model.

2.5.4 Generalizations

A common way to apply these models for fine-scale mapping with linkage disequilibrium is to consider a large population of normal chromosomes, within which there is a small but expanding population of disease chromosomes [36] [76]. The distinction between normal and disease chromosomes arises from the genetic nature of the disease: the disease chromosomes carry a faulty version of a gene. Since we can treat disease chromosomes as a separate population within a larger, stable population, we can focus our attention on modeling the behavior of the subpopulation.

The models must be modified to allow for genetic systems where the disease gene is linked to, and at disequilibrium with, a marker locus whose location in the genome is known. The underlying behavior of the disease population is typically taken to be as follows [36] [76]:

1. At some time in the past, $t = 0$, a single copy of the disease allele appeared in the population, on a chromosome that had a marker allele of type i.

2. Through the generations, the number of disease alleles in the population increased until the present time.

3. The distribution of marker alleles in the disease population changed over time through recombination events.

4. Mutation and selective pressures may also have had an impact.

The modifications to the models introduced above are relatively straightforward. We can modify the Moran model to account for growing populations. Also, we can make the populations described by branching processes grow by choosing offspring distributions with means greater than one. Modeling the recombination events between the disease and marker loci is possible within the Moran model through viewing the changes in marker allele to be a type of "mutation", the rate of which represents the recombination coefficient. This can be accomplished directly within the branching process framework by considering multi-type Galton-Watson models. In multi-type branching processes, one can explicitly model the transitions from one type, or in this case one marker allele, to another (see it e.g. [52]).

As these models form the basis for the results obtained in this thesis, Chapter 4 discusses them in greater detail.

Chapter 3

Linkage Disequilibrium Mapping

The power of genetic mapping via linkage disequilibrium lies in the fact that the disequilibrium between two loci decays at a rate that is influenced by the distance between them. Two questions arise before disequilibrium mapping can be used. First, disequilibrium must be present. Second, the mechanism that induced disequilibrium must be explained. The first issue can be addressed with the classical techniques of contingency table analysis. Some work has been done to distinguish between disequilibrium arising from various genetic phenomena. For instance, Chakraborty and Weiss [8] present a method that can be used to differentiate between disequilibrium induced through admixture of populations from that due to physical linkage of loci. Linkage analysis is also often used to explain linkage diseqiulibrium. If loci exhibit disequilibrium with a disease gene and are in a region of known linkage, then their allelic association is likely to be due to physical linkage.

Once the existence of linkage disequilibrium is verified, it can be used to localize disease genes within a background map of marker loci in one of two ways. The first option is to use the relative magnitudes of association to define a search region for the disease gene. The second possibility is to make assumptions about the population and use the resulting model to obtain a direct estimate of the recombination coefficient.

3.1 Simple Mapping

Geneticists first used linkage disequilibrium in a manner that has come to be called *Simple Disequilibrium Mapping* (see *e.g.* [11]). To use simple disequilibrium mapping, the researcher computes a measure of disequilibrium for each marker locus in the region of interest and plots the measures against the positions of the loci. The subregion that shows the greatest degree of disequilibrium becomes the first portion of DNA to be searched for the disease-influencing gene.

In order to utilize this technique, one must employ a measure of disequilibrium. We have already seen one possibility in the measure D from Equation 2.11. However, many measures of linkage disequilibrium exist. Some of them are much like D in that they treat the marker alleles as categories. Others rely on data from marker loci whose alleles can each be assigned a meaningful integer value.

3.1.1 Categorical Measures

The simplest categorical measures of allelic association are for two loci, each with two alleles. The measure D is one of these. Devlin and Risch [11] studied the behavior of the most commonly used simple disequilibrium measures of this type. These measures are listed in Table 3.2, using the notation defined in Table 3.1.

Note that each of the measures may be written as different scalings of D. This is true for D', however, only when the alleles are arranged in the table so that D is positive and the disease is rare relative to the associated marker allele frequency [70]. The actual formula is

$$D' = \begin{cases} \frac{p_{11}p_{22}-p_{12}p_{21}}{\min(p_{1\cdot}p_{\cdot 2},p_{\cdot 1}p_{2\cdot})} & \text{if } \; D > 0, \\ \frac{p_{11}p_{22}-p_{12}p_{21}}{\min(p_{\cdot 1}p_{1\cdot},p_{2\cdot}p_{\cdot 2})} & \text{otherwise.} \end{cases} \tag{3.1}$$

Devlin and Risch [11] conclude that the best of these measures of disequilibirum is δ. They arrive at this conclusion after studying the their behavior through time, letting D_t represent the value of D at time t. By assuming that the relative frequency of the disease allele remains constant for all t and that when $t = 0$ a single marker allele is associated with the disease allele, they conclude that the best estimate of D_0 is $p_{1\cdot}p_{22}$. This allows them to exploit the relationship shown in Equation 2.13 and conclude that

$$\delta = \frac{p_{11}p_{22} - p_{12}p_{21}}{p_{1\cdot}p_{22}} = \frac{D_t}{D_0} = (1-r)^t, \tag{3.2}$$

	M$_1$	M$_2$	...	M$_k$	Totals
Disease	p_{11}	p_{12}	...	p_{1k}	$p_{1\cdot}$
Normal	p_{21}	p_{22}	...	p_{2k}	$p_{2\cdot}$
	$p_{\cdot 1}$	$p_{\cdot 2}$	...	$p_{\cdot k}$	1

Table 3.1 Notation for joint and marginal allele frequencies from a $2 \times k$ table.

Measure	Formula
Δ	$\frac{p_{11}p_{22}-p_{12}p_{21}}{(p_{1\cdot}p_{2\cdot}p_{\cdot 1}p_{\cdot 2})^{1/2}}$
D'	$\frac{p_{11}p_{22}-p_{12}p_{21}}{p_{1\cdot}p_{\cdot 2}}$
δ	$\frac{p_{11}p_{22}-p_{12}p_{21}}{p_{1\cdot}p_{22}}$
d	$\frac{p_{11}p_{22}-p_{12}p_{21}}{p_{1\cdot}p_{2\cdot}}$
Q	$\frac{p_{11}p_{22}-p_{12}p_{21}}{p_{11}p_{22}+p_{12}p_{21}}$

Note: This formulation of D' is a special case discussed in the text.

Table 3.2 Disequilibrium measures commonly used for two bi-allelic loci.

This result indicates that δ depends directly on the recombination coefficient. Simulation experiments support this result in the sense that δ is the least dependent on other factors such as marker allele frequencies. While Devlin and Risch [11] conclude that δ is the best available measure of linkage disequilibrium, they also note that the behavior of D' is comparable.

The measures in Table 3.2 are appropriate only for two bi-allelic loci. As such, they are clearly insufficient for use with commonly used polymorphic loci. Hedrick [26], Morton and Wu [55] and Karlin and Piazza [38] studied measures of disequilibrium that may be used for markers with an arbitrary number of alleles. The simplest of these are scalings of the term

$$D^2 \;=\; \sum_{i=1}^{k}\sum_{j=1}^{l}\left(p_{ij}-p_{i\cdot}p_{\cdot j}\right)^2, \tag{3.3}$$

a generalization of D. Chakraborty *et al.* [7] note that there are at least five different classes of measures of disequilibrium for multiallelic loci. Each of these classes is formulated with respect to different functions of the haplotype frequencies, and each possesses distinct analytical properties.

None of the measures that have been proposed to date are entirely satisfactory. The primary concern is that they all are influenced by marginal allele frequencies. Devlin and Risch claim that the expected behavior of δ and D' do not depend on marker allele frequencies, and Hedrick makes the same claim about a generalization of D'. However, Lewontin [48] demonstrates that, while the limits of D are determined by the marginal allele frequencies due to the constraint that the joint allele frequencies be positive, D itself is indeterminate given a change in marginal allele frequencies due to the fact that there is only one degree of freedom in a 2×2 table. Since D itself is indeterminate, no measure that is a function of D and the marginal allele frequencies can be invariant to changes in the marginal allele frequencies.

This difficulty, along with the problem that evolutionary forces other than recombination can influence disequilibrium, has led some to try to measure associations from familial data. There are two methods for this. They are called *Haplotype Relative Risk* (HRR) [14] and the *Transmission Disequilibrium Test* (TDT) [65]. The aim of these family-based measures of association is to utilize genetic information from pedigrees to obtain estimates of the probabilities in Table 3.1 that will produce measures of disequilibrium that reflect only linkage, and not sampling or population effects.

In the case of HRR, one attempts to calculate the risk of contracting a disease, given the presence of a specific marker allele in linkage with the disease gene, relative to the risk of disease in the group that does not have that allele. For the case where there are two marker alleles, this is formulated as [74]

$$\mathrm{RR} \;=\; \frac{\frac{p_{11}}{p_{\cdot 1}}}{1-\frac{p_{11}}{p_{\cdot 1}}}\;\frac{1-\frac{p12}{p_{\cdot 2}}}{\frac{p_{12}}{p_{\cdot 2}}} \;=\; \frac{p_{11}}{p_{\cdot 1}-p_{11}}\;\frac{p_{\cdot 2}-p_{12}}{p_{12}} \;=\; \frac{p_{11}\;p_{22}}{p_{12}\;p_{21}}. \tag{3.4}$$

The p_{ij} are estimated as though individuals were randomly selected from the population, or that the disease and normal samples were obtained in the same manner, which is rarely the case. In order to overcome this, Falk and Rubenstein propose the HRR statistic as an adaptation of the RR statistic, with modifications on how the p_{ij} are estimated. As an illustration of how this is done, consider a recessive disease completely linked to a known marker locus. If we can unambiguously identify the disease/marker haplotype that was passed on from each parent to the affected offspring, the parental haplotypes not transmitted by the parents can be viewed as a random sample from the population of such haplotypes. Thus, we can use the non-transmitted parental haplotypes as the sample from which we estimate the p_{2j} and allele frequencies. Likewise, we can use the haplotypes transmitted to the affected offspring to estimate the p_{1j}.

In the original formulation, the behavior of the HRR was only known for systems where the probability of recombination was zero. Further modifications have been shown to be informative even when r is greater than zero [44]. Also, by taking into account the non-independence of transmitted and non-transmitted parental marker alleles, Knapp *et al.* [44] obtained the standard error of the estimator of HRR.

The TDT is quite similar to the HRR. It also uses familial information to construct the control sample, ensuring that the disease sample is compared against the proper population. The purpose of the TDT is to test for linkage between a marker and a disease locus that show population association [65]. The test is formulated by considering the $2n$ parents of n affected individuals. The parents are then classified by the joint behavior of the transmission of their marker alleles to their child. For example, consider the situation where there are two marker alleles, M_1 and M_2. We then classify the parent's transmission of alleles. The data

	Non-transmitted		
Transmitted	M_1	M_2	Total
M_1	a	b	$a+b$
M_2	c	d	$c+d$
Total	$a+c$	$b+d$	$2n$

Table 3.3 Transmitted and non-transmitted alleles from $2n$ parents for the transmission-disequilibrium test.

from such considerations can be tabulated as in Table 3.3, where, for example, b represents the number of parents that passed on an M_1 allele and did not pass on a M_2 allele to the affected child. The TDT test statistic is then defined as McNemar's test statistic

$$\chi^2_{tdt} = \frac{(b-c)^2}{(b+c)}, \tag{3.5}$$

which follows the χ^2 distribution with one degree of freedom (see *e.g.* [3]).

The families eligible for use with this method consist of at least one affected offspring and one heterozygous parent. Statistical properties of the test built from evaluation of the transmission of alleles from heterozygous parents to affected offspring were first considered by Spielman, McGinnis and Ewens [65], who found that it is a valid test for linkage in the presence of population association. Further considerations have been made. For example, the test can be generalized to the case when there are more then one affected offspring [65]. Others extended the TDT to marker loci with more than two alleles [64].

Both TDT and HRR are valid tests for association, and in the case of TDT, even linkage. Therefore they provide measures of linkage disequilibrium. Because they are family-based, they are not of primary interest for this work.

3.1.2 Moment Measures

General Markers

It is also possible to derive moment estimators that are related to population association. As one example, consider a sample of n gametes which have been sequenced at a given locus. Then, if we let k_{ij} be the number of sites at which gametes i and j differ within the locus, we can define an estimate of the variance of the number of site differences between pairs of sequences in the sample as

$$S_k^2 = \frac{1}{n^2}\sum_i^n\sum_j^n\left(k_{ij} - \frac{\sum_{i=1}^n\sum_{j=1}^n k_{ij}}{n^2}\right)^2 \tag{3.6}$$

In 1968, Sved [66] suggested that it should be possible to use S_k^2 as a measure of multilocus association. More than a decade later, Brown *et al.* [5] were able to express the estimator as a function of the pairwise linkage disequilibria between the sites. Chakraborty [9] [10] utilized the distribution of the number of heterozygous loci in an individual to further study the population associations. The results of these studies indicated that variations of S_k^2 could be used to test for nonrandom association of alleles.

Using the results of Brown *et al.*, [5] Hudson [31] derived an estimator of the parameter $4Nr$, where N is the effective population size. Using p_{ji} to denote the sample frequency of the i^{th} allele at site j, $h_j = 1 - \sum p_{ji}^2$ to denote the sample estimate of heterozygosity at site j and μ to denote the mutation rate, he showed that

$$\mathrm{E}[S_k^2 - \sum_j h_j + \sum_j h_j^2] = 4N\mu g(4Nr, n), \tag{3.7}$$

where $g(4Nr, n)$ is of the form

$$g(4Nr, n) \;=\; \frac{2}{(4Nr)^2} \int_0^c f(z)(4Nr - z)dz. \tag{3.8}$$

This equation led Hudson [31] suggest that an estimator for $4Nr$ could be the value that satisfies Equation 3.7, with sample values replacing the parameters. Wakeley [73] later proposed a modification in which he used

$$S_\pi^2 \;=\; \frac{2}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(k_{ij} \;-\; \frac{2}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} k_{ij} \right)^2 \tag{3.9}$$

in place of S_k^2 to obtain an estimating equation similar to that in Equation 3.7. Both Hudson [31] and Wakeley [73] verified that their estimates were valid via simulations. Wakeley [73] demonstrated that his estimator was somewhat better than that of Hudson [31].

The value of r estimated by these measures represents the average recombination rate per base pair within a length of DNA for which the nucleotide sequence is known for each homolog in the sample. Hence, these measures do not seem to be useful for genetic mapping. While it may be possible to modify these procedures to make them applicable to the problem of genetic mapping, we do not pursue that possibility in this thesis.

Microsatellite Markers

Another possible approach involves deriving moment measures for explicit use with microsatellite loci. Microsatellite loci belong to a broader class of Variable Number of Tandem Repeat (VNTR) loci [57]. These loci consist of tandem repeats of short DNA motifs, and their alleles are characterized by the number of observed repeats. Specifically, microsatellites consist of repeats of one to five nucleotides. For example, the sequence CAGCAG consists of two copies of the trinucleotide sequence CAG.

Microsatellite loci have come to be widely used in genetic mapping studies for several reasons. First, the human genome contains a large number of such loci [19] [20] [33]. Second, microsatellite loci usually exhibit a high number of alleles and are highly polymorphic, due to their high mutation rates [57] [25]. Third, it is reasonable to assume that many of these loci are not affected by selective pressures [34].

Another benefit of this class of loci is the fact that their alleles can be described by nonnegative integers. This simplifies mathematical modeling as alleles can be labeled with a meaningful integer: the number of repeats.

Kimmel *et al.* [40] take advantage of the properties of microsatellite loci and propose a model of their evolution. The model addresses the joint evolution of two microsatellite loci in two individuals drawn at random from a population. Using this population model, the authors derive the following moment measure

$$\kappa = \frac{Cov[(X_1 - Y_1)^2, (X_2 - Y_2)^2]}{Var[X_1]Var[X_2]},$$

where $[X_1,\ X_2]$ and $[Y_1,\ Y_2]$ are the haplotypes of two individuals. The expected value of κ is equal to $2 - r$ when the population model is evaluated at large values of the age parameter. For smaller values of the age parameter, κ's expected value is one.

Application of this measure to data from several tightly linked microsatellite loci resulted in estimates of κ that were close to one. Hence, other measures are required to be able to achieve estimates of r for microsatellite loci. Also, as measures from this model are for two microsatellite loci, it is inappropriate to use them to estimate the recombination rate between one microsatellite locus and a disease locus with discrete alleles.

3.2 Maximum Likelihood Mapping

The traditional approach to disequilibrium mapping is to choose some measure of disequilibrium, using it to identify regions in the genome demonstrating the greatest amount of allelic association. While it is possible to modify this simple mapping approach to obtain point estimates of the recombination fraction between a given marker locus and the unknown disease gene, the simple modifications are incapable of providing confidence bounds for the estimate. Hästbacka *et al.* [24]

took a first step by applying the classical methods of mutation assay initially proposed by Luria and Delbrück [50] to the propagation of a disease mutation within a population of chromosomes. Using Luria-Delbrück methodology, they were able to obtain both point estimates and approximate confidence limits for the recombination coefficient between a marker locus and a disease gene.

Another large class of methods has been proposed since the work by Hästbacka *et al.* [24]. This class consists of applying maximum likelihood methods to sampled data via various population models. The remainder of this section will discuss the use of maximum likelihood concepts in genetic mapping using linkage disequilibrium data. It will first present methods for single markers and conclude with techniques relating to the use of more than one marker.

3.2.1 Single Marker

Several methods for likelihood-based linkage disequilibrium mapping exist. These methods fall into two classes. The techniques from the first class develop likelihoods that are based on linkage disequilibrium measures. Those in the second class develop likelihoods for r directly by modeling the behavior of the disease population. This subsection will describe both types of methods.

Hill and Weir [29] were the first to try to utilize likelihood methods to map disease genes with linkage disequilibrium data. They devised a method by modeling the measure Δ^2 (see Table 3.2) through the parameter , $= 4Nr$, where N is the effective population size. This choice was based on the fact that Δ^2 is a function of , , rather than r alone (see *e.g.* [28]) The reason for this is that Δ^2 decays as recombination events occur through the history of a population, and the number of recombination events in a population depends on the size of the population: a large population has more recombination events than a small one. Hill and Weir modeled this relationship through formulae for the probability density functions of the haplotype counts, conditional on , as well as other genetic effects, such as selection. Simulation studies showed that that the resulting likelihood is not highly influenced by the degree of selection at the loci. However, their results also indicated that their likelihood for r did not peak sharply for values of r near the maximum. This suggested that linkage disequilibrium may not provide precise genetic maps.

The second likelihood method to be based on a measure of linkage disequilibrium was proposed by Terwilliger [68]. The primary purpose of his method was to provide a single degree of freedom test for genetic markers with k alleles. Rather than calculating the Pearson chi-squared statistic from a $2 \times k$ contingency table, Terwilliger made the assumption that a single unknown marker allele was associated with the disease. This allowed him to use a parameter, λ, to represent the proportion by which the associated allele is increased over its population frequency.

Using this formulation, he constructed the likelihood

$$\begin{aligned} L \;=\; & \sum_{i=1}^{k} [p_{i\cdot}\{p_{i\cdot} + \lambda(1-p_{i\cdot})\}^{n_{1i}}\ \{p_{i\cdot} - \lambda(1-p_{i\cdot})p_D/(1-p_D)\}^{n_{2i}} \\ & \times \prod_{j \neq i} \{p_{j\cdot}(1-\lambda)\}^{n_{1j}}\ \{p_{j\cdot} + \lambda p_{j\cdot} p_D/(1-p_D)\}^{n_{2j}}], \end{aligned} \tag{3.10}$$

where p_D is the population frequency of the disease, and the p_{ij} and n_{ij} represent the allele frequencies and sampled allele counts, following the notation established in Table 3.1.

This likelihood is essentially the product of the multinomial likelihoods obtained for the disease and normal samples. To observe this, note that if we assume the i^{th} marker allele to be associated with the disease allele, then $p_{i\cdot} + \lambda(1-p_{i\cdot})$ is the proportion of the disease chromosomes that contain the associated allele, $p_{i\cdot} - \lambda(1-p_{i\cdot})p_D/(1-p_D)$ is the proportion of normal chromosomes containing the associated allele, $p_{j\cdot}(1-\lambda)$ is the proportion of disease chromosomes containing the j^{th} (unassociated) allele and $p_{j\cdot} + \lambda p_{j\cdot} p_D/(1-p_D)$ is the proportion of normal chromosomes that contain the j^{th} (unassociated) allele. Since we do not know which allele is associated, the full likelihood is the weighted sum of the likelihoods where a single allele is fixed as the one associated with the disease. This likelihood can be utilized to construct a test for association. Simulations show that the resulting likelihood ratio test is more powerful than the standard Pearson statistic against the alternative that a single allele is associated with the disease [68]. Once association has been established, one may use the formula for the decay of linkage disequilibrium (Equation 2.13) to define the relationship $\lambda = \alpha(1-r)^t$, where

α is the initial degree of linkage disequilibrium. This formulation then yields an estimate of the recombination fraction.

Devlin *et al.* [12] propose a third approach, using the measure δ in Table 3.2. Since $-(1/t)\log\delta$ is approximately equal to r in large populations [27], they modeled the random variable $Y = -\log\delta$ for likelihood mapping purposes. They relied on simulation results to assert that the distribution of Y could be approximated with a Gamma density. This made it possible to model not only mean behavior for a marker locus, but also provided a method to allow for variability in Y.

The authors obtained a likelihood using the relationship between Y and r mentioned above, and assuming that Y followed a Gamma density. They recommended that the likelihood be calculated along a grid of recombination fractions to obtain the maximum likelihood estimate.

A final class of likelihood-based method disequilibrium methods is due to Kaplan *et al.* [36]. Their technique involves modeling the evolution of a small disease population within a large non-disease population. Under this scenario, one may consider the distribution of marker alleles for normal and disease populations separately, with that of the disease population being of primary interest since the normal population is assumed to be large. From this setup, a sample from the disease population can be modeled with a multinomial distribution if one assumes the marker allele frequencies in the disease population to be known. Hence, we can define the log likelihood with the multinomial sampling model:

$$l(r|\, p_{1d}, p_{2d}, \ldots, p_{kd}) = \kappa + \sum_{i=1}^{k} n_{id} \log\left(p_{id}\right), \tag{3.11}$$

where p_{id} and n_{id} represent the allele frequencies and sampled allele counts from the disease population. Note that the p_{id} sum to one, unlike the p_{1i} in Table 3.1. Note also that this is a function of r since the recombination coefficient influences the marker allele frequencies. Because the marker allele frequencies are random variables reflecting the evolutionary history of the disease population, the log likelihood for r is given by,

$$l(r) = \kappa + \mathrm{E}\left[\sum_{i=1}^{k} n_{id} \log\left(p_{id}\right)\right], \tag{3.12}$$

where the expectation must be evaluated over the entire history of the population.

Solving this expectation becomes of primary importance. The first step is to model the evolution of the p_{id} with some population genetic model. Kaplan *et al.* [36] do this with a Galton-Watson branching process with Poisson offspring distributions. They then exploit the properties of the Poisson distribution, namely that the sum of independent Poisson distributions is another Poisson distribution, to obtain the stochastic recursion

$$X_i(t+1) \sim \mathrm{Poisson}\{(1+\lambda)[(1-r)X_i(t) + rX_T(t)p_{in}]\}, \tag{3.13}$$

where $X_i(t)$ is the number of disease chromosomes with the i^{th} marker allele in the t^{th} generation, $X_T(t)$ is the total number of disease chromosomes in the

t^{th} generation and λ is the growth parameter. They then propose a simulation scheme that enables them to estimate the likelihood via Monte Carlo simulations by assuming that the disease mutation initially occurred on a chromosome with a specified marker allele.

Xiong and Guo [76] take a different approach to solve the expectation in Equation 3.12. They use differential equations to approximate the first- and second-order moments of the Wright-Fisher model in what is known as a diffusion approximation [17]. In particular, they show that the first two moments of p_{id} satisfy the following differential equations if there are k marker alleles:

$$\frac{d\mathrm{E}[p_{id}(t)]}{dt} = \mathrm{E}[g_i(t)], \qquad i = 1, 2, \ldots, k; \tag{3.14}$$

$$\frac{d\mathrm{E}[p_{id}^2(t)]}{dt} = \mathrm{E}\left[\frac{p_{id}(t)\{1 - p_{id}(t)\}}{X_T(t)}\right] + 2\mathrm{E}[g_i(t)p_{id}(t)], \quad i = 1, 2, \ldots, k; \tag{3.15}$$

and

$$\frac{d\mathrm{E}[p_{id}(t)p_{jd}(t)]}{dt} = -\mathrm{E}\left[\frac{p_{id}(t)p_{jd}(t)}{X_T(t)}\right] + \mathrm{E}[g_i(t)p_{jd}(t)] + \mathrm{E}[g_j(t)p_{id}(t)], \quad i \neq j, \tag{3.16}$$

where $g_i(t) = \mathrm{E}[p_{id}(t+1) - p_{id}(t)|p_{1d}(t), p_{2d}(t), \ldots, p_{kd}(t)]$. Given a population genetic model for the p_{id}, it is possible to solve the equations for these moments either analytically or numerically. For example, if we assume that $p_{1d} = 1$ and $p_{jd} = 0, j = 2, 3, \ldots k$ and that there is no mutation at either the marker or disease locus and also that the marker allele frequencies in the normal population (p_{in})

are fixed, we solve to find that

$$\begin{aligned} \mathrm{E}[p_{1d}(t)] &= e^{-rt} + (1-e^{-rt})p_{1n}, \\ \mathrm{E}[p_{jd}(t)] &= (1-e^{-rt})p_{jn}, \quad j = 2, 3, \ldots k. \end{aligned} \tag{3.17}$$

Using these first- and second-order moments, Xiong and Guo [76] construct approximations to the likelihood based on truncated Taylor series expansions about the mean of the process. The first-order approximation (FOA) is

$$l_1(r) \approx \kappa + \sum_{i=1}^{k} n_{id} \log\left(\mathrm{E}\left[p_{id}(t)\right]\right), \tag{3.18}$$

and the second-order approximation (SOA) is given by

$$\begin{aligned} l_2(r) &\approx \kappa + \textstyle\sum_{i=1}^{k} n_{id} \log\left(\mathrm{E}\left[p_{id}(t)\right]\right) \; + \; \nabla\ \mathrm{E}[p(t) - \pi(t)] \\ &\qquad + \tfrac{1}{2}\mathrm{E}\left\{[p(t)-\pi(t)]H[p(t)-\pi(t)]^T\right\} \\ &= \kappa + \textstyle\sum_{i=1}^{k} n_{id} \log\left(\mathrm{E}\left[p_{id}(t)\right]\right) \; + \; \tfrac{1}{2}\{tr[HM(t)] - \pi(t)H\pi(t)^T\}, \end{aligned} \tag{3.19}$$

where $p(t) = (p_{1d}(t)\ p_{2d}(t)\ \ldots\ p_{kd}(t))$, $\pi(t) = (\mathrm{E}[p_{1d}(t)]\ \mathrm{E}[p_{2d}(t)]\ \ldots\ \mathrm{E}[p_{kd}(t)])$ and

$$\nabla \;=\; \frac{\partial}{\partial p_{id}(t)} \sum_{j=1}^{k} n_{jd} \log\left(p_{jd}(t)\right)\Bigg|_{p(t)=\pi(t)} \tag{3.20}$$

are row vectors, and $M(t) = \mathrm{E}[p(t)p(t)^T]$ and

$$H \;=\; \frac{\partial^2}{\partial p_{id}(t)\partial p_{jd}(t)} \sum_{l=1}^{k} n_{ld} \log\left(p_{ld}(t)\right)\Bigg|_{p(t)=\pi(t)} \tag{3.21}$$

are square matrices. We do not need to evaluate the gradient, as its term cancels from the equation, but performing the differentiation for the Hessian, we find that it is a diagonal matrix with the diagonal elements equal to

$$H_{ii} \;=\; \frac{-n_{id}}{\mathrm{E}\left[p_{id}(t)\right]^2}. \qquad i = 1, 2, \ldots, k, \tag{3.22}$$

Both the simulation method of Kaplan *et al.* [36] and the approximation approach of Xiong and Guo [76] are capable of providing point estimates and confidence intervals for the recombination coefficient. However, there are several drawbacks to each method. The branching process model used by Kaplan *et al.* [36] is simplistic; it does not account for mutation, either at the marker locus or at the disease locus. Also, their Monte Carlo method is subject to simulation variability, which can be significant due to the exponential growth of supercritical branching processes. Decreasing the effect of simulation error through replication can be prohibitive due to the amount of time required to simulate the likelihood. The method of Xiong and Guo [76] relies on a diffusion approximation to make a discrete-time model work in continuous time. It also uses the assumption that the frequency of the disease allele remains constant in the population, which can be problematic if the population is not large. Other difficulties arise when applying their methodology. Specifying the form of the model in terms of the differential equations for the moments can be demanding, as can obtaining the solutions of those equations.

3.2.2 Multiple Markers

If the data from more than one marker is available, it can provide great improvement in mapping a disease gene. However, all of the methods presented in the previous section are for the case when there is a single marker allele. This raises the question of the extendibility of these techniques to more than one marker.

The methods of Kaplan *et al.* [36] and Xiong and Guo [76] rely on specific population genetic models. Hence, they can be modified to accommodate several markers. In fact, both groups have proposed models dealing with the joint evolution of two marker loci. However, modeling haplotype data in this way becomes combinatorially complex, as one must account for the possibility of recombination events in many intervals. Another difficulty that arises with modeling multiple marker loci is that the likelihoods used with these models are based on haplotype counts, and there is currently little published haplotype data. The difficulties that arise in models for many markers have led most researchers to rely on what are called composite, or pseudo, likelihoods [12] [68] [76].

The composite likelihood was first applied to the analysis of spatial data by Besag [2], who called it a pseudo-likelihood. The term "composite" prevailed because the log-likelihood is a composition, actually a sum, of marginal or conditional log likelihoods. For example, if X_1 and X_2 are two dependent random variables, then the complete log-likelihood could be written as $\log f(x_1;\theta) + \log f(x_2|x_1;\theta)$, where θ represents the parameters in the model. This leads to two possible composite likelihoods when either one of other of the components is difficult to obtain: $cl = \log f(x_1;\theta) + \log f(x_2;\theta)$ and $cl = \log f(x_1|x_2;\theta) + \log f(x_2|x_1;\theta)$. Note that the terms in the sum need not be independent, an important consideration when mapping a disease gene within a map of tightly linked marker loci.

Because composite log likelihoods are sums of ordinary, or conditional, log likelihoods they retain many of the properties of classical log likelihoods. For instance, since the Kullback-Liebler information inequality holds for each of the component log-likelihoods, it also does for the sum:

$$\mathrm{E}_{\beta_0}[l_i(\beta)] \leq \mathrm{E}_{\beta_0}[l_i(\beta_0)] \Rightarrow \sup_{\beta} \mathrm{E}_{\beta_0}[cl(\beta)] = \mathrm{E}_{\beta_0}[cl(\beta_0)]. \tag{3.23}$$

Hence, composite likelihoods can be shown to provide consistent estimators under additional assumptions about their convergence [49].

There are two additional arguments for the use of composite likelihoods. First, they provide a method of estimation when the full likelihood is difficult to specify. Second, they often represent the parts of the model about which we have the most “knowledge”. This means that we either have more data for the component likelihoods than for the full likelihood, or we are more confident in our model for the component likelihoods, or both. Because of these issues, composite likelihood methods are thus far the method of choice in linkage disequilibrium mapping.

3.3 Prospectus

Application of these methods to several diseases for which the disease gene has been identified has met with mixed success. All of the methods provide reasonable search regions for the genes causing Cystic Fibrosis and Diastrophic Dysplasia, based on the maps published by Kerem *et al.* [39] and Hästbacka *et al.* [23], respectively. However, the simpler methods do not perform satisfactorily with other diseases, such as Huntington's disease and Friedrich Ataxia [36] [76], which has lead some researchers to conclude that the usefulness of linkage disequilibrium mapping may be limited for diseases that are genetically complex [12] [36] [43] [51]. Xiong and Guo have provided some evidence that this is not the case by demonstrating that the use of appropriate population genetic models makes it possible to map the genes even of complex diseases.

All of the applications discussed in the previous section have their drawbacks. For example, the methods of Terwilliger [68] and Devlin *et al.* [12] are formulated purely in terms of linkage disequilibrium measures and not in terms of the assumed population model. They are therefore difficult to modify to allow for different population scenarios. The diffusion approximation of Xiong and Guo, as described previously, is unwieldy due to complicated differential equations. In addition to this limitation, it is only an approximation to the assumed Wright-Fisher model. Also, the model of Kaplan *et al.* [36] is very simplified and requires time-consuming simulation.

The remainder of this thesis provides a framework to map disease genes via linkage disequilibrium by expanding the work of Kaplan *et al.* [36] and Xiong and Guo [76]. It utilizes the theory of branching processes to obtain approximations to likelihoods for r. It also obtains similar results for a time-continuous version of the Moran model via the coalescent. After obtaining the approximations to the likelihood ignoring all factors other than recombination, it proceeds to generalize the model, considering the case where there are two markers and allowing for mutation both at the marker loci and at the disease locus. It will also address issues arising when data from more than one marker locus are available.

Chapter 4

Population Models

Chapter 2 provided a brief introduction to three population models and their potential for use in fine-scale mapping via linkage disequilibrium. This chapter more fully explores multi-type Galton-Watson branching processes and Moran/Coalescent processes. The chapter will focus on two general aspects of each model: their descriptions and their first two moments.

4.1 Descriptions

This section outlines the population models used through the remainder of this thesis. It first outlines the basic properties of these models for populations whose elements are all alike. It then expands the basic notions from these models to the case where there are several different types of elements in the populations. Throughout the discussion, recall that the elements in the populations being modeled are chromosomes carrying a disease mutation. The distinguishing feature that separates the chromosomes into several classes is the marker allele present on each chromosome.

4.1.1 The Galton-Watson Process

The information presented below is well known. As such, the discussion of this section will not contain specific references. However, the majority of the material was obtained from texts written by Harris [21], Mode [52] and Resnick [62].

The simplest Galton-Watson branching process considers a population of objects evolving in discrete time. That is, it tracks the size of the population in successive generations, not the actual ancestry or the specific times that individual objects are born. Hence the focus is the number of objects present in the t^{th} generation, which we denote by $X(t)$. The behavior of $X(t)$ as time progresses is governed by the basic assumptions underlying the Galton-Watson process.

The fundamental concept that drives the evolution of a Galton-Watson branching process is the notion that all of the elements in the population are alike. Two explicit assumptions are required to describe this idea. The first is that the elements in the process all produce offspring according to the same probability law. The second is that each of the elements in the population produce offspring independently of one another.

The consequence of this model is that the sequence consisting of the number of elements in the population at each generation, $X(1)$, $X(2)$, $\ldots$, forms a Markov chain. That is, the probability law that governs the number of elements in generation $(t+1)$ depends only on the number of elements in generation t. In fact, given

$X(t)$, the number of particles in the next generation can be written as

$$X(t+1) \;=\; \sum_{i=1}^{X(t)} x_{it}, \tag{4.1}$$

where x_{it} is a realization of the random variable X_{it}, the number of offspring produced by the i^{th} element from the t^{th} generation. Since the offspring distributions are independent and identically distributed for all $i \geq 1$ and $t \geq 1$, they can be described by a single *probability generating function* (pgf). This pgf can be written as

$$f(s) \;=\; \sum_{k=0}^{\infty} p_k s^k, \qquad |s| \leq 1, \tag{4.2}$$

where p_k is the probability that a element will produce k offspring, and s is a complex variable. Note that the expected number of offspring is given by

$$\mathrm{E}[X_{it}] \;=\; \frac{\mathrm{d}}{\mathrm{d}s} f(s)\bigg|_{s=1} \;=\; \mu, \qquad i = 1, \ldots, X(t),\; t = 1, 2, \ldots \tag{4.3}$$

and the variance is

$$\mathrm{Var}[X_{it}] \;=\; \frac{\mathrm{d}^2}{\mathrm{d}s^2} f(s)\bigg|_{s=1} + \mu - \mu^2 \;=\; \sigma^2, \qquad i = 1, \ldots, X(t),\; t = 1, 2, \ldots. \tag{4.4}$$

The pgf of $X(t+1)$ is derived by forming functional iterates of Equation 4.2, or

$$f_{t+1}(s) \;=\; f\left[f_t(s)\right], \qquad t = 1, 2, 3, \ldots, \tag{4.5}$$

where $f_1(s) = f(s)$. This form arises from the fact that the number of particles in the population at time $t+1$ is the sum of offspring from a random number of particles. While explicit expressions for such objects seldom exist, they can still be used to study the process. Specifically, they make it possible to compute moments of the number of elements in the population at any time t. For instance, if a single element existed in the population at time $t = 0$, the first two moments are

$$\mathrm{E}[X(t)] \;=\; \mu^t, \tag{4.6}$$

and

$$\mathrm{Var}[X(t)] \;=\; \begin{cases} \frac{\sigma^2\mu^t(\mu^t-1)}{\mu^2-\mu}, & \mu \neq 1, \\ t\sigma^2, & \mu = 1. \end{cases} \tag{4.7}$$

These moments are obtained by applying Equations 4.3 and 4.4 to the iterated pgf in Equation 4.5.

These results are not directly applicable to the problem of genetic mapping, as they apply only to populations consisting of identical elements. In order to use Galton-Watson processes for linkage disequilibrium mapping, it is necessary to consider models for non-identical particles. This is so because disease chromosomes differ through the possession of one of k distinct marker alleles. Therefore, the discussion now turns to multi-type branching process models.

Like the simple Galton-Watson process, the multi-type version focuses on the number of elements in the population in each generation. The distinction between the two processes is that the multi-type process must track the number of elements in each class. For instance, the Galton-Watson process with one class of elements is capable of modeling the number of disease chromosomes in successive generations. The multi-type process can track not only the total number of disease chromosomes, but also the number of disease chromosomes carrying each of k specific marker alleles. Like the process with one state, the multi-type process is a Markov process. However, its states are vectors, $\mathbf{X(t)}$, whose components are nonnegative integers, $X_i(t)$ for $i \in \{1, 2, \ldots, k\}$.

As with the Galton-Watson process of a single type, the offspring distributions can be defined with probability generating functions (pgfs). In this case, the pgfs must be vector-valued. For instance, the pgf for the offspring distribution of a single chromosome with the i^{th} marker allele is

$$\begin{aligned} f^i(s_1, s_2, \ldots, s_k) &= \sum_{n_1,\ldots,n_k}^{\infty} p^i(n_1, n_2, \ldots, n_k) s_1^{n_1} s_2^{n_2} \ldots s_k^{n_k}, \\ & \qquad\qquad |s_1|, |s_2|, \ldots, |s_k| \leq 1, \end{aligned} \tag{4.8}$$

where $p^i(n_1, n_2, \ldots, n_k)$ is the probability that an element of type i has n_1 children of type 1, n_2 children of type 2, and so on up to n_k children of type k. The probability generating function of $\mathbf{X(t)}$ is obtained through forming functional iterates of the component offspring distributions. In particular, the pgf for elements of type i is

$$f^i_{t+1}(\mathbf{s}) \;=\; f^i\left[f^1_t(\mathbf{s}), f^2_t(\mathbf{s}), \ldots, f^k_t(\mathbf{s})\right], \qquad t = 1, 2, 3, \ldots. \tag{4.9}$$

As with the branching process of a single type, specification of the pgfs makes it possible to obtain expressions for the moments of the vector-valued distributions generated by multi-type processes. The moments obtained from single-type

Galton-Watson branching processes are scalars. Those of multi-type branching processes are matrices, whose elements depend on both the parental and offspring type. For instance, the elements of these matrices, μ_{ij}, represent the expected number of offspring of type j from a parent of type i. This quantity can be written mathematically for the case where the parent is an element of type i in generation zero as

$$\mu_{ij} \;=\; \mathrm{E}[X_j(1)|\mathbf{X}(0) = \mathbf{e}_i] \;=\; \frac{\partial}{\partial s_j} f^i(\mathbf{s})\bigg|_{\mathbf{s}=\mathbf{1}}, \qquad i, j = 1, 2, \ldots, k, \tag{4.10}$$

where $\mathbf{e}_i$ is a vector whose i^{th} entry is a one, and whose other components are zero. The matrix of first moments is then defined as $\mathbf{M} = (\mu_{ij})$. A vector generalization of Equation 4.6 results in

$$\mathrm{E}[\mathbf{X}(t)|\mathbf{X}(0)] \;=\; \mathbf{X}(0)\mathbf{M}^t, \qquad t = 0, 1, 2, \ldots, \tag{4.11}$$

provided that all μ_{ij} are finite and that they are not all zero.

To obtain the second moments, define $\mathbf{C}(t)$ to be the matrix whose $(i, j)^{\text{th}}$ element is $\mathrm{E}[X_i(t)X_j(t)|\mathbf{X}(0)]$. Then considering the conditional expectations $\mathrm{E}[X_i(t+1)X_j(t+1)|\mathbf{X}(t)]$, provides the result that

$$\mathbf{C}(t+1) \;=\; \mathbf{M}^T\mathbf{C}(t)\mathbf{M} + \sum_{i=1}^{k} \mathbf{V}_i \mathrm{E}[X_i(t)|\mathbf{X}(0)], \qquad t = 0, 1, 2, \ldots. \tag{4.12}$$

This assumes that the entries in the one-generation covariance matrices,

$$\begin{aligned} \mathbf{V}_{i\,[j,l]} \;=\;& \mathrm{E}[X_j(1)X_l(1)|\mathbf{X}(0) = \mathbf{e}_i] \\ & -\mathrm{E}[X_j(1)|\mathbf{X}(0) = \mathbf{e}_i]\;\mathrm{E}[X_l(1)|\mathbf{X}(0) = \mathbf{e}_i], \quad i, j, l = 1, 2, \ldots, k, \end{aligned} \tag{4.13}$$

are all finite. Note that $\mathbf{V}_i$ is simply the covariance matrix of the number of offspring produced by a single element of type i. Applying Equation 4.12 repeatedly, gives

$$\begin{aligned}\mathbf{C}(t) &= \left(\mathbf{M}^T\right)^t \mathbf{C}(0)\mathbf{M}^t \\ &+ \sum_{j=1}^{t} \left(\mathbf{M}^T\right)^{t-j} \left\{\sum_{i=1}^{k} \mathbf{V}_i \mathrm{E}[X_i(j-1)|\mathbf{X}(0)]\right\} \mathbf{M}^{t-j}, \qquad t = 1, 2, \ldots.\end{aligned} \tag{4.14}$$

With this framework to obtain the expected counts and covariances, it is now possible to formulate specific branching process models for the propagation of disease genes within a population. Estimates of these moments can then be used to form approximate maximum likelihood estimators for the location of a disease gene.

4.1.2 The Moran/Coalescent Process

The discussion now returns to the Moran model [53], which was briefly introduced in Chapter 2. This model considers a population of haploid chromosomes evolving in continuous time. In the simplest case, this population is of a constant size. However, the population size can be generalized to describe various forms of deterministic population growth or decline, where the population size at time t is modeled by the function $N(t)$. What follows is a description of a version of the Moran model for the case where the population size is constant, and consists of $N(t)$ disease chromosomes which may contain one of k possible alleles at a locus near the disease gene. The description requires the specification of three population behaviors: birth/death, recombination and mutation.

- *Birth/Death*: When a chromosome leaves the population, it is immediately replaced by another chromosome. The epochs of each chromosome's death and rebirth constitute a homogeneous Poisson process with an intensity of one "generation" on $[0, \infty)$. Hence, a chromosome's lifetime is exponentially distributed with a mean of one generation.

- *Recombination*:

  – Moran Model: At times of birth/death, the offspring chromosome is sampled from the pool of chromosomes with the same marker allele with probability $1 - r$ (no recombination). The offspring chromosome is sampled from the entire population with probability r (recombination occurs).

  – Coalescent Version: The time lines of the chromsomes contain time points where recombinations between the marker and disease loci occur. These recombination epochs constitute independent homogeneous Poisson processes on $[0, \infty)$ with intensities equal to r.

- *Mutation*: The lifetimes of the chromosomes contain epochs where mutations occur at the marker locus. These epochs form independent homogeneous

Poisson processes on $[0, \infty)$, with intensities ν. When a mutation occurs, a marker allele of type j replaces one of type i with probability ν_{ij}.

The birth/death, recombination and mutation processes of each chromosome are assumed to be mutually independent.

Forward equations can be written to describe the behavior of the population process. These forward equations are based on the behavior of the process as it proceeds through a infinitesimal time interval, conditioning on the process being in a given state prior to the time interval. The result of using this approach is that the behavior of the population can be studied through differential equations, as can be done with any time-continuous Markov chain.

However, as noted previously, the Moran model can be directly studied by running time backwards [13]. This makes it possible to utilize the coalescent to simplify the process of obtaining the results. To use the coalescent with the Moran model, it is necessary to superimpose the basic behaviors of the model on the construct of a coalescent. What is needed is to proceed as with any with time-continuous Markov chains, and obtain the generator matrix, otherwise known as the matrix of intensities, **Q**. This matrix defines the behavior of recombination and mutation such that the transition probabilities among marker alleles are obtained through the operation

$$\mathbf{P}(t) \;=\; e^{\mathbf{Q}t}. \tag{4.15}$$

These transition probabilities can then be superimposed on a coalescent process with branch lengths that are independent and exponentially distributed.

Figure 4.1 provides a schematic representation of this coalescent process. This figure represents two disease chromosomes, one with marker allele j and the other with marker allele l, sampled at time t. Under the Moran/Coalescent model, these two chromosomes are descended from an ancestral chromosome that existed at time $t - \tau$ in the past, where τ is an exponentially distributed random variable with mean proportional to the population size, $N(t)$. If this ancestral chromosome

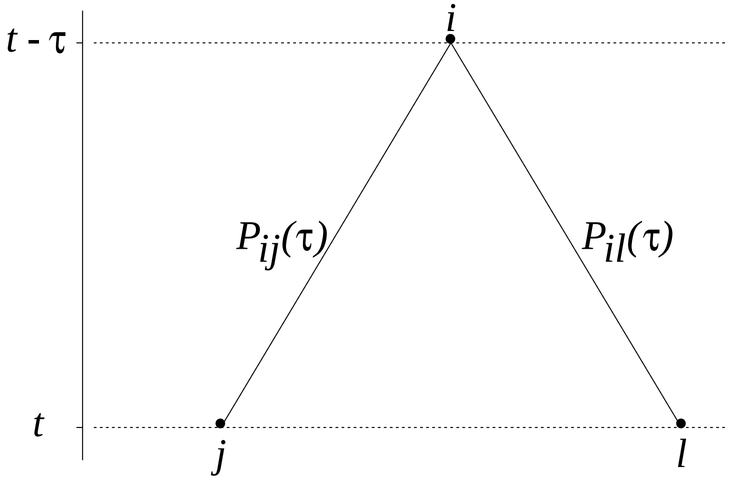

Figure 4.1 Representation of a coalescent tree for a sample of two chromosomes.

was of type i, then for j and l to be its descendents, they must have either undergone mutation or recombination events. The quantities $P_{ij}(\tau)$ and $P_{il}(\tau)$ represent the probabilities that a disease chromosome of type i will transit, through recombination or mutation, to a chromosome of type j or l, respectively, in a time interval of length τ.

As with the Galton-Watson process, the first two moments of the distribution of marker allele frequencies, given initial conditions and other assumptions can be used to form approximations to the likelihood for r. In this case, let $p(t)$ be a row vector containing the relative frequencies of the marker alleles in the disease population at time t. By assuming the form of $p(0)$, the first moments are obtained from the transition probabilities through the relationship

$$\pi(t) \;=\; \mathrm{E}[p(t)] \;=\; p(0)\mathbf{P}(t). \tag{4.16}$$

Obtaining the second moments is more involved. First, let

$$R_{jk}(t) = \Pr[X_1 = j, X_2 = k], \tag{4.17}$$

where X_1 and X_2 are randomly selected chromosomes. If the common ancestor of X_1 and X_2 was type i and lived at a fixed time, τ units in the past, then independence assumptions give that

$$R_{jk}(t)|_{\tau,i} \;=\; P_{ij}(\tau)P_{ik}(\tau). \tag{4.18}$$

Since i and τ are not fixed, but random quantities, the conditioning needs to be removed by applying the law of total probability to obtain

$$R_{jk}(t) \;=\; \int_0^{\infty} \sum_{i=1}^{m} \pi_i(t-\tau) P_{ij}(\tau) P_{ik}(\tau) \frac{1}{N(t-\tau)} e^{-\int_0^{\tau} \frac{du}{N(t-u)}}\, d\tau, \tag{4.19}$$

or after a change of variables

$$R_{jk}(t) \;=\; \int_{-\infty}^{t} \sum_{i=1}^{m} \pi_i(\sigma) P_{ij}(t-\sigma) P_{ik}(t-\sigma) \frac{1}{N(\sigma)} e^{-\int_{\sigma}^{t} \frac{du}{N(u)}}\, d\sigma. \tag{4.20}$$

In matrix form, this is

$$\mathbf{R}(t) \;=\; \int_{-\infty}^{t} \mathbf{P}^T(t-\sigma)\; \Pi(\sigma)\; \mathbf{P}(t-\sigma)\; \frac{1}{N(\sigma)} e^{-\int_{\sigma}^{t} \frac{du}{N(u)}}\, d\sigma, \tag{4.21}$$

where $\Pi(t) = diag[\pi(t)]$ is the diagonalized expected distribution. Separating out the initial conditions provides the expression

$$\begin{aligned}
\mathbf{R}(t) \;&=\; \textstyle\int_{-\infty}^{0} \mathbf{P}^T(t)\mathbf{P}^T(-\sigma)\Pi(\sigma)\mathbf{P}(-\sigma)\mathbf{P}(t)\frac{1}{N(\sigma)} e^{-\int_{\sigma}^{0} \frac{du}{N(u)}}\; e^{-\int_{0}^{t} \frac{du}{N(u)}}\, d\sigma \\
&\quad+\; \textstyle\int_0^t \mathbf{P}^T(t-\sigma)\; \Pi(\sigma)\; \mathbf{P}(t-\sigma)\; \frac{1}{N(\sigma)} e^{-\int_{\sigma}^{t} \frac{du}{N(u)}}\, d\sigma \\
&=\; \mathbf{P}^T(t)\; \mathbf{R}(0)\; \mathbf{P}(t)\; e^{-\int_0^t \frac{du}{N(u)}} \\
&\quad+\; \textstyle\int_0^t \mathbf{P}^T(t-\sigma)\; \Pi(\sigma)\; \mathbf{P}(t-\sigma)\; \frac{1}{N(\sigma)} e^{-\int_{\sigma}^{t} \frac{du}{N(u)}}\, d\sigma.
\end{aligned} \tag{4.22}$$

The solution of the integral in Equation 4.22 yields only the joint probabilities of marker allele sharing for two randomly sampled disease chromosomes. More work must be done to obtain the covariances.

Equation 4.16 contains the expected probabilities of sampling a disease chromosome with a given marker allele are contained in. To obtain the second moments, consider $\mathrm{E}[p_i^2(t)]$ and $\mathrm{E}[p_i(t)p_j(t)]$, where i and j index specific marker alleles. The population contains $N(t)$ elements of distinct types, with their relative frequencies given by the vector $p(t)$. This information can be used to obtain estimates of the second-order moments. Namely, examining all chromosomes from the population, $X_i(t), i \in \{1, 2, \ldots, N(t)\}$, and letting $I_i(X_j)$ be the indicator function for disease chromosome X_j having a marker allele of type i, produces the result that

$$\begin{aligned} \mathrm{E}[p_i^2(t)] &= \mathrm{E}\left[\tfrac{1}{N(t)}\textstyle\sum_{j=1}^{N(t)} I_i(X_j)\ \tfrac{1}{N(t)}\textstyle\sum_{k=1}^{N(t)} I_i(X_k)\right] \\ &= \tfrac{1}{N^2(t)}\textstyle\sum_{j=1}^{N(t)}\ \textstyle\sum_{k=1}^{N(t)} \mathrm{E}\left[I_i(X_j) I_i(X_k)\right] \\ &= \tfrac{1}{N^2(t)}\textstyle\sum_{j=1}^{N(t)} \mathrm{E}\left[I_i(X_j)\right] \ + \ \tfrac{1}{N^2(t)}\displaystyle\sum_{j\neq k} \mathrm{E}\left[I_i(X_j) I_i(X_k)\right] \\ &= \tfrac{1}{N(t)}\ p_i(t) \ + \ \tfrac{N(t)-1}{N(t)}\ R_{ii}(t), \end{aligned} \tag{4.23}$$

and

$$\begin{aligned} \mathrm{E}[p_i(t)\ p_j(t)] &= \mathrm{E}\left[\tfrac{1}{N(t)}\textstyle\sum_{k=1}^{N(t)} I_i(X_k)\ \tfrac{1}{N(t)}\textstyle\sum_{l=1}^{N(t)} I_j(X_l)\right] \\ &= \tfrac{1}{N^2(t)}\textstyle\sum_{k=1}^{N(t)} \mathrm{E}\left[I_i(X_k) I_j(X_k)\right] \ + \ \tfrac{1}{N^2(t)}\displaystyle\sum_{k\neq l} \mathrm{E}\left[I_i(X_k) I_j(X_l)\right] \\ &= \tfrac{N(t)-1}{N(t)}\ R_{ij}(t), \end{aligned} \tag{4.24}$$

where $R_{ij}(t)$ represents the $[i, j]^{\mathrm{th}}$ entry in $\mathbf{R}(t)$ (see Equation 4.22). Therefore, the entries in the covariance matrix for this version of the Moran model are given by

$$\mathrm{Var}[p_i(t)] \ = \ R_{ii}(t) \ - \ p_i^2(t) \ + \ \frac{1}{N(t)}\left[p_i(t) - R_{ii}(t)\right], \tag{4.25}$$

and

$$\mathrm{Cov}[p_i(t), p_j(t)] \ = \ R_{ij}(t) \ - \ p_i(t)\ p_j(t) \ - \ \frac{1}{N(t)} R_{ij}(t). \tag{4.26}$$

It is now possible to form specific models, based on various assumptions concerning the behavior of the marker alleles in the population of disease chromosomes. These models provide methods that allow for the estimation of the moments of the marker allele frequencies. The moments, in turn, make it possible to form likelihoods that can be used to estimate the location of a disease gene relative to a marker locus.

4.2 Descriptions of Specific Models

Past sections provided information about the first two moments of two distinct classes of models. This section discusses specific forms of each model that can be used for fine-scale mapping. It first describes several global assumptions required by the population models and concludes with the specfication of a variety of models useful for linkage disequilibrium mapping.

The assumptions required by both the Galton-Watson and Moran/Coalescent models are:

1. The population of normal chromosomes is larger than the subpopulation carrying a disease mutation.

2. The frequencies of the alleles at the linked marker are fixed quantities in the general population.

3. At some time in the past, a single disease chromosome existed in the entire population.

4. The ancestral disease chromosome carried the marker allele that is the most common marker allele in the disease population at present.

5. The marker locus is linked to the disease locus, with a recombination rate of r, with $0 \leq r \leq 0.5$.

6. The subpopulation of disease chromosomes is growing exponentially, at a rate of $1 + \lambda$, with $\lambda > 0$.

7. Chromosomes pair at random during the reproductive process.

Assumption 2 is not overly restrictive, since the allele frequencies in a large population change slowly with time, especially relative to those in a much smaller population (see it e.g. [76]). Assumption 4 is due to the fact that the most frequent disease allele is most likely to be the ancestral allele.

	M_1	M_2	$\ldots$	M_k	Totals
Disease	p_{1d}	p_{2d}	$\ldots$	p_{kd}	1
Normal	p_{1n}	p_{2n}	$\ldots$	p_{kn}	1

Table 4.1 Notation for marker allele frequencies in populations of disease and normal chromosomes.

This paragraph establishes some notation. Table 4.1 contains the notation for the marker allele frequencies used hereafter for both the disease and normal populations. Equation 4.27 displays the mutation matrix, **U**, where

$$\mathbf{U} = \begin{pmatrix} \nu_{11} & \nu_{12} & \cdots & \nu_{1k} \\ \nu_{21} & \nu_{22} & \cdots & \nu_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ \nu_{k1} & \nu_{k2} & \cdots & \nu_{kk} \end{pmatrix}, \tag{4.27}$$

and ν_{ij} equals the probability that marker allele i mutates to marker allele j in one generation. Note that ν_{ii} is the probability that the i^{th} marker allele does not mutate. The constant ν_d represents the probability that a new disease mutation occurs during the reproductive process. For ease of notation, define the matrix P as

$$P = \begin{pmatrix} p_{1n} & p_{2n} & \cdots & p_{kn} \\ p_{1n} & p_{2n} & \cdots & p_{kn} \\ \vdots & \vdots & & \vdots \\ p_{1n} & p_{2n} & \cdots & p_{kn} \end{pmatrix}. \tag{4.28}$$

With the notation redefined, it is now appropriate to discuss models for the joint evolution of a marker locus and a disease locus. The next section will focus on models for multi-type branching processes. The final section in the chapter will discuss models based on the time-continuous Moran model. Three types of models are of interest. The first model considers only recombination. It is the most tractable, but the least realistic. The second allows for mutation at the marker locus. The third model allows for mutations at both the marker and disease loci.

4.2.1 Galton-Watson Processes

This section outlines the systems of probability generating functions that describe the behavior of disease chromosomes, when a disease gene is linked to a marker locus. The discussion includes descriptions of the properties the models should possess, and how these features are incorporated into probability generating functions.

Recall that the branching processes in use here are branching processes with independent Poisson offspring distributions, which serve as approximations to the Wright-Fisher population model. Because the offspring distributions are Poisson, they are completelly specified by their mean behavior. The remainder of this section describes the desired average behavior of the branching process offspring distributions, and incorporates these features in explicit systems of probability generating functions

Recombination Only

Here it is assumed that two phenomena uniquely govern the average behavior of the offspring distributions: transitions from one allele to another through recombination and the growth rate of the disease. As mentioned earlier, the population of disease chromosomes exhibits an exponential rate of growth on average, with a growth rate of $1 + \lambda$. Modeling the recombination process is somewhat more involved. Its description is presented below.

Assumptions 1 through 7, supply enough information to derive the marker allele transitions. Consider the situation where a disease chromosome with marker allele i gives birth to a disease chromosome with marker allele j. If $i \neq j$, the transition probability is equal to the probability that the disease chromosome recombines with a chromosome with a marker allele of type j, or rp_{jn}. Likewise, the probability that a disease chromosome with marker allele i gives birth to a disease chromosome with the same marker allele is equal to $1 - r + rp_{in}$, since the chromosome either does not recombine or it recombines with a chromosome carrying the same marker allele. This transition matrix can be compactly expressed as $[(1-r)\mathbf{I} + rP]$, where $\mathbf{I}$ is a $k \times k$ identity matrix and P is defined in Equation 4.28.

This transition matrix assumes that a disease chromosome produces a single offspring chromsome. Multiplying the transition matrix by the growth rate provides a description of the matrix of first moments with the desired behavior. This matrix is

$$\mathbf{M}_r = (1+\lambda)\left[(1-r)\mathbf{I} + rP\right]. \qquad (4.29)$$

This matrix can now be used to define the offspring distributions.

The mean matrix in Equation 4.29 defines the system of pgfs for this model as

$$f^i(\mathbf{s}) \;=\; e^{(1+\lambda)(1-r+rp_{in})(s_i-1)} \prod_{j\neq i} e^{(1+\lambda)rp_{jn}(s_j-1)}, \qquad i=1,2,\ldots,k, \qquad (4.30)$$

where $\mathbf{s}$ is a vector containing $s_1, s_2, \ldots, s_k$. We can verify that this system of pgfs has the specified mean behavior by differentiating, and evaluating at $\mathbf{s} = \mathbf{1}$ as in Equation 4.10. The system of pgfs also yeilds the single-generation covariance matrices as defined in Equation 4.13. Because the offspring process have independent Poisson distributions, the variances are equal to the mean and the covariances are equal to zero. Hence, if $\mathbf{X}(0) = \mathbf{e}_i$, the covariance matrix is

$$\mathbf{V}_{\mathbf{r}i} \;=\; diag[\mathbf{e}_i\, \mathbf{M}_r], \qquad i=1,2,\ldots,k. \qquad (4.31)$$

When mutation rates are low, this model may suffice. However, allowing for mutations to occur at the marker locus should provide additional mapping power. This is especially true if the marker loci have a relatively high probability of mutation, as do at least one class of commonly-used markers: microsatellite loci.

Recombination plus Mutations at the Marker Locus

Accommodating mutations at the marker locus first requires the same basic assumptions needed in the case where only recombination is allowed. Therefore, it is possible to expand the recombination-only model to allow for mutations at the marker locus. In addition to the assumptions required by the recombination-only model, this model assumes that mutation events occur independently of recombination events, with the probabilities of mutating from one allele to another contained in the mutation matrix $\mathbf{U}$, as listed in Equation 4.27.

One issue remains in defining the process including mutations. Do mutations occur before or after recombination? Mutations are most likely to occur in the process of DNA replication, and replication precedes recombination, it seems plausible that mutations take place prior to recombination. This is the case studied in most detail. However, for completeness, both cases are condisered here.

If mutations occur before recombination, and mutations and recombinations are independent, then the joint recombination/mutation matrix is the product of the mutation and recombination transition matrices. Multiplying the transition matrix by the growth rate yields the matrix of first moments

$$\begin{aligned} \mathbf{M}_{mr} &= (1+\lambda)\mathbf{U}\left[(1-r)\mathbf{I} + rP\right] \\ &= (1+\lambda)\left[(1-r)\mathbf{U} + r\mathbf{U}P\right] \\ &= (1+\lambda)\left[(1-r)\mathbf{U} + rP\right]. \end{aligned} \tag{4.32}$$

The simplification arises because the rows of $\mathbf{U}$ sum to one and the entries in the columns of P are all equal. The result also has a reasonable biological explanation: if no recombination event occurs, then a mutation event will persist in the offspring. However, if a mutation event is followed by recombination between the disease and marker loci, then the mutation is lost from the disease chromosome and replaced by a randomly chosen marker allele.

Modeling this mean behavior with Poisson offspring distributions leads to the system of generating functions

$$f^i(\mathbf{s}) \;=\; \prod_{j=1}^{k} e^{(1+\lambda)(1-r)\nu_{ij}(s_j-1)}\; e^{(1+\lambda)rp_{jn}(s_j-1)}, \qquad i=1,2,\ldots,k. \tag{4.33}$$

This system of pgfs has the mean behavior specified in Equation 4.32. Also, the covariance matrices are similar to those in Equation 4.31, namely

$$\mathbf{V}_{\mathbf{mr}i} \;=\; diag[\mathbf{e}_i\, \mathbf{M}_{mr}]. \tag{4.34}$$

Furthermore, note that if there is no mutation ($\mathbf{U} = \mathbf{I}$), then the pgfs reduce to those given in Equation 4.30.

If the mutation mechanism operates after recombination, then the moment matrix is

$$\begin{aligned}\mathbf{M}_{rm} \;&=\; (1+\lambda)\left[(1-r)\mathbf{I} + rP\right]\mathbf{U} \\ &=\; (1+\lambda)\left[(1-r)\mathbf{U} + rP\mathbf{U}\right].\end{aligned} \tag{4.35}$$

In this case, there is no simplification. In fact,

$$P\mathbf{U} = \begin{pmatrix} \sum_{i=1}^{k} p_{in}\nu_{i1} & \sum_{i=1}^{k} p_{in}\nu_{i2} & \cdots & \sum_{i=1}^{k} p_{kn}\nu_{ik} \\ \sum_{i=1}^{k} p_{in}\nu_{i1} & \sum_{i=1}^{k} p_{in}\nu_{i2} & \cdots & \sum_{i=1}^{k} p_{kn}\nu_{ik} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{k} p_{in}\nu_{i1} & \sum_{i=1}^{k} p_{in}\nu_{i2} & \cdots & \sum_{i=1}^{k} p_{kn}\nu_{ik} \end{pmatrix}. \tag{4.36}$$

This leads to the system of pgfs

$$\begin{aligned} f^i(\mathbf{s}) \;&=\; \prod_{j=1}^{k} \exp\left[(1+\lambda)(1-r)\nu_{ij}(s_j-1)\right] \\ &\quad\times \exp\left[(1+\lambda)r\textstyle\sum_{l=1}^{k} p_{ln}\nu_{lj}(s_j-1)\right], \qquad i=1,2,\ldots,k. \end{aligned} \tag{4.37}$$

As with the pgfs in Equation 4.33,

$$\mathbf{V}_{\mathbf{rm}i} = diag[\mathbf{e}_i \, \mathbf{M}_{rm}], \tag{4.38}$$

and if $\mathbf{U} = \mathbf{I}$ then the system reduces to Equation 4.30.

As mutations at marker alleles are typically more common than new disease mutations, this model should be reasonable for a variety of diseases. However, there may be cases where it is of interest, and even necessary, to allow for the possibility of mutations at the disease locus.

Recombination plus Mutations at the Marker and Disease Loci

Allowing for mutations at the disease locus is a simple extension of the previous models. The updated model adds a new mutation process that is independent of the recombination and marker mutation processes. This mutation mechanism adds new disease alleles at a rate of ν_d on a randomly chosen disease chromosome.

Adding this process to the model where mutation occurs prior to recombination, defines the moment matrix

$$\begin{aligned} \mathbf{M}_{mrd} &= (1+\lambda)\left[(1-r)\mathbf{U} + rP + \mu_d P\right] \\ &= (1+\lambda)\left[(1-r)\mathbf{U} + (r+\mu_d)P\right]. \end{aligned} \tag{4.39}$$

This leads to the Poisson system of pgfs

$$f^i(\mathbf{s}) = \prod_{j=1}^{k} e^{(1+\lambda)(1-r)\mu_{ij}(s_j-1)} \; e^{(1+\lambda)(r+\mu_d)p_{jn}(s_j-1)}, \qquad i = 1, 2, \ldots, k, \tag{4.40}$$

whose matrix of first moments is given in Equation 4.39 and whose one-step covariance matrices are

$$\mathbf{V}_{\mathbf{mrd}i} \;=\; diag[\mathbf{e}_i\,\mathbf{M}_{mrd}]. \tag{4.41}$$

This system models recombination between a disease and a marker locus in tandem with possible mutations at either the marker or the disease loci, or at both of them. The simplified models are recovered by setting appropriate mutation terms to zero.

The pgfs above define a variety of evolutionary models: recombination only, recombination with mutations at the marker locus and recombination with mutations possible at both the marker and disease loci. These probability generating functions, and their functional iterates define the behavior of our branching processes and will be used in the next chapter to derive estimators of the recombination fraction.

4.2.2 The Moran/Coalescent Model

This section describes the generator matrices that define the time-continuous Markov chains corresponding to various versions of the Moran model. Because the recombination probabilities can be viewed as mutation intensities, the off-diagonal entries of the generator matrices are equal to the transition probabilities. Since the row sums of generator, or intensity, matrices must be zero (see it e.g. [62]), the diagonal elements negate the sum of the off-diagonal entries.

The same assumptions for the recombination and mutation processes apply to both the branching process models and the Moran/Coalescent models. Therefore, the off-diagonal entries in the generator matrices are equal to the off-diagonal entries in the transition matrices presented in the previous section. The discussion below describes these generator matrices for the three specific cases discusses previously.

Recombination Only

When transitions are only possible through recombination, the elements of the intensity matrix come from the matrix $\mathbf{M}_r$ in Equation 4.29 when $\lambda = 0$. In this case, the off-diagonal entries are exactly the off-diagonal elements of the matrix rP. Therefore, the generator matrix is

$$\mathbf{Q}_r \;=\; r \begin{pmatrix} p_{1n} - 1 & p_{2n} & \dots & p_{kn} \\ p_{1n} & p_{2n} - 1 & \dots & p_{kn} \\ \vdots & \vdots & \ddots & \vdots \\ p_{1n} & p_{2n} & \dots & p_{kn} - 1 \end{pmatrix} \;=\; r\,[P - \mathbf{I}\,]\,. \tag{4.42}$$

Recombination plus Mutations at the Marker Locus

If marker mutations occur prior to recombination events, the the matrix of intensities can be obtained from entries in $\mathbf{M}_{mr}/(1+\lambda)$ from Equation 4.32. This leads to the generator matrix

$$\mathbf{Q}_{mr} \;=\; (1-r)\mathbf{U} + rP - \mathbf{I}. \tag{4.43}$$

Likewise, if mutation occurs after recombination, modifying Equation 4.35 yields the matrix of intensities

$$\mathbf{Q}_{rm} \;=\; (1-r)\mathbf{U} + rP\mathbf{U} - \mathbf{I}. \tag{4.44}$$

Recombination plus Mutations at the Marker and Disease Loci

Again, the intensity matrix for the Moran/Coalescent model can be obtained from the transition matrix for the Galton-Watson model. In this case, the quantities are shown in Equation 4.39. Placing the transition probabilities in the off-diagonal elements and subtracting the row sums from the diagonal entries produces the generator matrix:

$$\mathbf{Q}_{mrd} \;=\; (1-r)\mathbf{U} + (r+\mu_d)P - (1+\mu_d)\mathbf{I}. \tag{4.45}$$

The intensity matrices for these different mechanisms of recombination and mutation are sufficient to characterize the transition laws of the associated Markov chains. These transition laws, along with a model for population growth $(N(t))$ provide the information we need to obtain the first two moments listed in Equations 4.16, 4.25 and 4.26. These moments will then make it possible to obtain approximate maximum likelihood estimators of r, which is done in the next chapter.

Chapter 5

Mapping a Disease Gene with One Marker

The last chapter described models of population behavior to be employed in linkage disequilibrium mapping. This chapter uses the moments from each model to obtain approximate maximum likelihood estimates of the recombination fraction between a marker locus and a putative disease locus. It first discusses estimators based on multi-type Galton-Watson branching processes. The chapter concludes by obtaining estimates of r from variations of the time-continuous Moran model. For each class of models, there are three models of interest: recombination only, recombination plus marker mutations and recombination plus marker and disease mutations.

5.1 The Galton-Watson Model

Three pieces of information are needed to formulate estimators based on a Galton-Watson branching process: the first and second moments and the Hessian as defined in Equations 4.11, 4.14 and 3.21 respectively. These quantities make it possible to calculate the log likelihoods written in Equations 3.18 and 3.19. The log likelihoods then provide point and interval estimates of the recombination coefficient.

The form of the Hessian matrix remains constant regardless of the population model. What differs from model to model are the first two moments of the marker allele frequencies in the disease population. The sections that follow describe the moments for the three models of interest, with the additional assumption that the original disease chromosome possessed the marker allele that is currently the most common in the disease population. If the marker alleles are labeled by their frequency in the disease population (i.e. allele 1 is the most frequent allele), this assumption becomes

$$\mathbf{X}(0) \;=\; \mathbf{e}_1. \tag{5.1}$$

These moments, once obtained, are applied to construct likelihoods for the estimation of r.

The moments obtained for the branching process models represent means and covariances of marker allele counts, rather than frequencies. As such, the appropriate sampling model is the hypergeometric. The appropriate likelihood should therefore be

$$L(r) = \mathrm{E}\left[\frac{\left(\sum_{j=1}^{k} n_{jd}\right)!\left(\sum_{j=1}^{k} X_{jd} - \sum_{j=1}^{k} n_{jd}\right)!\prod_{i=1}^{k} X_{id}!}{\left(\sum_{j=1}^{k} X_{jd}\right)!\prod_{i=1}^{k} n_{id}!(X_{id} - n_{id})!}\right], \tag{5.2}$$

rather than the multinomial model as presented in Equation 3.12. However, the multinomial likelihood is a good approximation to the hypergeometric distribution if the population of disease chromosomes is reasonably large (see it e.g. [15]).

The multinomial likelihood brings with it several computational conveniences. First, the approximations to the expected frequencies, such as

$$\mathrm{E}[p_{id}] \;\approx\; \frac{\mathrm{E}[X_{jd}]}{\sum_{j=1}^{k} \mathrm{E}[X_{jd}]}, \qquad i = 1, 2, \ldots, k, \tag{5.3}$$

are more tractable than approximations to factorial moments like

$$\mathrm{E}[X_{id}!] \;\approx\; \mathrm{E}[X_{id}]!. \tag{5.4}$$

Also, it is convenient to avoid difficulties that one encounters when working with factorials. This is especially convenient as it is necessary to differentiate the likelihoods, not only to find maxima, but to refine approximations.

5.1.1 Recombination Only

This section builds the first- and second-order approximations to the likelihood for r. The first-order approximation depends on the matrix in Equation 4.29 and the second-order approximation depends on the matrices defined in Equation 4.31.

First Order Approximation

Making the assumption stated in Equation 5.1, and assuming that the disease mutation occurred t generations in the past, makes it possible to obtain the first moments of the process as described in Equation 4.11. Because $P^t = P$ (see

Equation 4.28) and $\mathbf{I}^t = \mathbf{I}$,

$$\begin{aligned}
\mathbf{M}_r^t/(1+\lambda)^t &= [(1-r)\mathbf{I} + rP]^t \\
&= r^tP^t + \textstyle\sum_{i=1}^{t-1} \frac{t!}{i!(t-i)!}\{(1-r)\mathbf{I}\}^i\{rP\}^{t-i} + (1-r)^t\mathbf{I}^t \\
&= r^tP + \textstyle\sum_{i=1}^{t-1} \frac{t!}{i!(t-i)!}(1-r)^i r^{t-i}P + (1-r)^t\mathbf{I} \qquad (5.5) \\
&= r^tP + P\left[1 - r^t - (1-r)^t\right] + (1-r)^t\mathbf{I} \\
&= (1-r)^t\mathbf{I} + \left[1-(1-r)^t\right]P, \qquad t = 1, 2, \ldots.
\end{aligned}$$

The first-order approximation is determined by the quantities contained in the first row of $\mathbf{M}_r^t/(1+\lambda)^t$, or

$$\pi_r(t) \approx \Big(p_{1n} + (1-p_{1n})(1-r)^t \;\; p_{2n}(1-(1-r)^t) \;\; \ldots \;\; p_{kn}(1-(1-r)^t)\Big). \qquad (5.6)$$

The approximation to the likelihood is therefore

$$l(r) \;=\; n_{1d}\log\{p_{1d} + (1-p_{1d})(1-r)^t\} + \sum_{i=2}^{k} n_{id}\log\{p_{id}[1-(1-r)^t]\}. \qquad (5.7)$$

Differentiating the log-likelihood, and equating it to zero, yields the expression

$$\frac{n_d - n_{1d}}{1-(1-\hat{r})^t} \;=\; \frac{n_{1d}(1-p_{1n})}{p_{1n} + (1-p_{in})(1-\hat{r})^t}, \qquad (5.8)$$

where $n_d = \sum_{i=1}^k n_{id}$. The resulting point estimate is

$$\hat{r} \;=\; 1 - \left(\frac{p_{1d} - p_{1n}}{1-p_{1n}}\right)^{\frac{1}{t}}. \qquad (5.9)$$

This is a simple function of what has been defined as p_{excess} [47], which is equal to the measure of disequilibrium δ (see Table 3.2) when $k = 2$. This result indicates that the branching process approximation produces an estimate corresponding to a special case of the decay of disequilibrium noted in Equation 2.13. The values of r where the log-likelihood is two units less than the value at the maximum defines an approximate ninety-five percent confidence interval. A FORTRAN program that calculates these point and interval estimates is included in Appendix A.

Second Order Approximation

The first-order approximation was formed by taking approximations to the expected allele frequencies, based on the expected allele counts from the branching process model. This section derives the covariance matrix for the allele counts from the branching process. Using this covariance matrix and modifying the second-order likelihood from Equation 3.19, produces a higher-order approximation to the likelihood for r.

Equation 4.14 lists a quantity, $\mathbf{C}(t) = \mathrm{E}[\mathbf{X}(t)\mathbf{X}(t)^T]$, which is analogous to the matrix $M(t) = \mathrm{E}[p(t)p(t)^T]$ required to complete the second order approximation to the likelihood (see Equation 3.19). Finding the form of $\mathbf{C}(t)$ is done in three parts. The first is $\left(\mathbf{M}^T\right)^t \mathbf{C}(0)\mathbf{M}^t$. Note that $(\mathbf{M}^t)^T = \left(\mathbf{M}^T\right)^t$. This is true since bringing the transpose inside the parentheses reverses the order of multiplication of matrices that are all identical (see it e.g. [4]). Also,

$$\mathbf{C}(0) \;=\; \mathrm{E}[\mathbf{X(0)X(0)^T}] \;=\; \begin{pmatrix} 1 & 0 & \ldots & 0 \\ 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 \end{pmatrix}, \tag{5.10}$$

because of the assumption that $\mathbf{X(0)} \;=\; \mathbf{e_1}$. Therefore,

$$\left(\mathbf{M}_r^T\right)^t \mathbf{C}(0)\mathbf{M}_r^t \;=\; \left(\mathbf{e}_1\mathbf{M}_r^t\right)^T \left(\mathbf{e}_1\mathbf{M}_r^t\right), \tag{5.11}$$

where $\mu_r(t)$ is defined in Equation 5.6.

The second quantity is $SVE \;=\; \sum_{i=1}^{k} \mathbf{V}_i\mathrm{E}[X_i(j-1)|\mathbf{X}(0)]$. In this case, the form of $\mathbf{V}_i$ is defined in Equation 4.31 and $\mathrm{E}[X_i(j-1)|\mathbf{X}(0)]$ is the $(1,i)$ element of $\mathbf{M}_r^{j-1}$. Since the terms in the sum are diagonal matrices, multiplied by scalars, the resulting matrix is also diagonal. Considering the (m,m) element of the resulting sum, reveals that

$$\begin{aligned} SVE_{[m,m]} \;&=\; \sum_{i=1}^{k} \mathbf{M}_{[i,m]}\;\mathbf{M}_{[1,i]}^{j-1} \;=\; \sum_{i=1}^{k} \mathbf{M}_{[1,i]}^{j-1}\;\mathbf{M}_{[i,m]} \\ &=\; \mathbf{M}_{[1,m]}^{j}. \end{aligned} \tag{5.12}$$

Therefore,

$$SVE \;=\; \sum_{i=1}^{k} \mathbf{V}_i\mathrm{E}[X_i(j-1)|\mathbf{X}(0)] \;=\; diag[\mathbf{e}_1\mathbf{M}^j]. \tag{5.13}$$

What remains is $B_j = \sum_{j=1}^{t} \left(\mathbf{M}^T\right)^{t-j} diag[\mathbf{e}_1\mathbf{M}_r^j]\ \mathbf{M}^{t-j}$. Performing the matrix multiplication, gives the result that

$$\begin{aligned}\frac{B_{j[1,1]}}{(1+\lambda)^{2t-j}} &= [p_{1n} + (1-p_{1n})(1-r)^{t-j}]^2[p_{1n} + (1-p_{1n})(1-r)^j] \\ &\quad + p_{1n}^2[1-(1-r)^{t-j}]^2\ [1-(1-r)^j] \sum_{l=2}^{k} p_{ln};\end{aligned} \tag{5.14}$$

$$\begin{aligned}\frac{B_{j[i,i]}}{(1+\lambda)^{2t-j}} &= p_{in}^2[1-(1-r)]^2[p_{1n} + (1-p_{in})(1-r)^j] \\ &\quad + p_{in}[p_{in} + (1-p_{in})(1-r)^{t-j}]^2[1-(1-r)^j] \\ &\quad + \{p_{in}^2[1-(1-r)^{t-j}]^2[1-(1-r)^j] \\ &\qquad \times \sum_{l\neq 1,i} p_{ln}\}, \qquad\qquad i = 2,\ldots,k;\end{aligned} \tag{5.15}$$

$$\begin{aligned}\frac{B_{j[1,i]}}{(1+\lambda)^{2t-j}} = \frac{B_{j[i,1]}}{(1+\lambda)^{2t-j}} &= p_{in}[p_{1n} + (1-p_{1n})(1-r)^{t-j}] \\ &\quad \times[p_{1n} + (1-p_{1n})(1-r)^j][1-(1-r)^{t-j}] \\ &\quad +\{p_{1n}p_{in}[1-(1-r)^{t-j}][1-(1-r)^j] \\ &\qquad \times[p_{in} + (1-p_{in})(1-r)^{t-j}]\} \\ &\quad +\{p_{1n}p_{in}[1-(1-r)^{t-j}]^2\ [1-(1-r)^j] \\ &\qquad \times \sum_{l\neq 1,i} p_{ln}\}, \qquad i = 2,\ldots,k;\end{aligned} \tag{5.16}$$

and

$$\begin{aligned}\frac{B_{j[l,i]}}{(1+\lambda)^{2t-j}} = \;& p_{in}p_{ln}[p_{1n} + (1-p_{1n})(1-r)^{j}][1-(1-r)^{t-j}]^2 \\ & + \{p_{in}p_{ln}[p_{in} + (1-p_{in})(1-r)^{t-j}] \\ & \quad \times[1-(1-r)^{j}][1-(1-r)^{t-j}]\} \\ & + \{p_{in}p_{ln}[p_{ln} + (1-p_{ln})(1-r)^{t-j}] \\ & \quad \times[1-(1-r)^{j}][1-(1-r)^{t-j}]\} \\ & + \{p_{in}p_{ln}\ [1-(1-r)^{t-j}]^2[1-(1-r)^{j}] \\ & \quad \times \sum_{m\neq 1,i,l} p_{mn}\}, \qquad i=2,\ldots,k;\ l=2,\ldots,k;\ i\neq l.\end{aligned} \tag{5.17}$$

Simplifying the expressions for $B_{j[i,l]}$, and summing j from 1 to t, produces the covariance matrix of the counts, $\mathbf{V}(t)$ because of the form of Equation 5.11. The entries in $\mathbf{V}(t)$ are:

$$\begin{aligned}\mathbf{V}_{[1,1]}(t) = \;& (1-p_{1n})\frac{[(1-r)(1+\lambda)]^{t}-[(1-r)(1+\lambda)]^{2t}}{r-\lambda(1-r)} \\ & + p_{1n}(1-p_{1n})\frac{[(1-r)(1+\lambda)]^{2t}-(1+\lambda)^{t}}{\lambda(1-r)^2-r(2-r)} \\ & + [2p_{1n}(1-p_{1n})(1-r)^{t} + p_{1n}^2]\frac{(1+\lambda)^{2t}-(1+\lambda)^{t}}{\lambda};\end{aligned} \tag{5.18}$$

$$\begin{aligned}\mathbf{V}_{[i,1]}(t) = \mathbf{V}_{[1,i]} = \;& [p_{1n}(1-2p_{in})(1-r)^{t} + p_{1n}p_{in}]\frac{(1+\lambda)^{2t}-(1+\lambda)^{t}}{\lambda} \\ & - p_{in}(1-2p_{1n})\frac{[(1-r)(1+\lambda)]^{t}-[(1-r)(1+\lambda)]^{2t}}{r-\lambda(1-r)} \\ & - p_{1n}p_{in}\frac{[(1-r)(1+\lambda)]^{2t}-(1+\lambda)^{t}}{\lambda(1-r)^2-r(2-r)}, \qquad i=2,\ldots,k;\end{aligned} \tag{5.19}$$

$$\begin{aligned}\mathbf{V}_{[i,i]}(t) \;=\;& p_{in}^2[1-2(1-r)^t]\tfrac{(1+\lambda)^{2t}-(1+\lambda)^t}{\lambda} \\ & -p_{in}(1-2p_{in})\tfrac{[(1-r)(1+\lambda)]^t-[(1-r)(1+\lambda)]^{2t}}{r-\lambda(1-r)} \\ & +p_{in}(1-p_{in})\tfrac{[(1-r)(1+\lambda)]^{2t}-(1+\lambda)^t}{\lambda(1-r)^2-r(2-r)}, \qquad i=2,\ldots,k;\end{aligned} \tag{5.20}$$

and

$$\begin{aligned}\mathbf{V}_{[i,l]}(t) \;=\;& p_{in}p_{il}[1-2(1-r)^t]\tfrac{(1+\lambda)^{2t}-(1+\lambda)^t}{\lambda} \\ & +2p_{in}p_{il}\tfrac{[(1-r)(1+\lambda)]^t-[(1-r)(1+\lambda)]^{2t}}{r-\lambda(1-r)} \\ & -p_{in}p_{il}\tfrac{[(1-r)(1+\lambda)]^{2t}-(1+\lambda)^t}{\lambda(1-r)^2-r(2-r)}, \\ & \qquad\qquad i=2,\ldots,k;\; l=2,\ldots,k;\; i\neq l.\end{aligned} \tag{5.21}$$

This covariance matrix is not in the correct scale to be directly useful in the formation of a second-order approximation to the likelihood for r. It is the covariance matrix of the disease allele counts while the terms in Equation 3.19 are for disease allele percentages. In order to overcome this drawback, it is possible to scale the elements in the likelihood so that they are of the same magnitude as the counts.

To perform this scaling, note that

$$\mathrm{E}[X_i(t)] \approx (1+\lambda)^t\mathrm{E}[p_{id}(t)] \tag{5.22}$$

(see Equations 5.3 and 5.5). Hence, multipling each element of $\pi(t)$ by $(1+\lambda)^t$ to produces $\mu(t)$. This has the effect of creating a different scaling of the likelihood, or

$$l(r) \;=\; \kappa^* + \sum_{i=1}^{k} n_{id}\log(\mu_i(t)) + \frac{1}{2}\left\{tr[H_x\mathbf{C}(t)] - \mu_i(t)H_x\mu_i(t)^T\right\}, \qquad (5.23)$$

where

$$H_{x[i,i]} \;=\; \frac{-n_{id}}{\mathrm{E}[X_i(t)]^2}, \qquad i = 1, \ldots, k. \qquad (5.24)$$

Using this likelihood, it is now possible to find second-order estimates of the recombination fraction. The computer program for the estimation of disease gene location with composite likelihoods found in Appendix A performs this estimation as part of its functionality.

5.1.2 Recombination plus Mutations at Marker and Disease Loci

In order to make the branching process model more realistic, consider the case where mutations may occur at the marker and disease loci. Note that the only difference between these two models is the parameter, μ_d, which models the rate of new disease "mutations" in the process (these mutations may include disease genes added through immigration, etc.). This section discusses first- and second-order approximations to the likelihood for r.

First Order Approximation

The matrix of first moments found in Equation 4.39 defines the mean behavior of the process. In order to find a closed-form solution for the first-order approximation to the likelihood, it is necessary to find an expression for $\mathbf{M}^t_{mrd}$. However, this matrix power depends directly on the powers of the mutation matrix, $\mathbf{U}$. It is possible to find powers of $\mathbf{U}$ for specific forms of mutation. However, in order to allow for general patterns of mutation, we utilize a computer program to calculate the matrix powers to obtain the first-order approximation. The FORTRAN program in Appendix A for composite likelihoods includes options that allow for first-order estimation with general mutation matrices and arbitrary rates of disease mutation, even for a single marker locus.

Second Order Approximation

As does the first-order approximation, the second-order approximation relies in the ability to obtain a general form of the powers of the moment matrix, $\mathbf{M}_{mrd}^{t}$. Again, in order to retain functionality for general marker mutation patterns, we use a computer program to calculate the powers of the mean matrix and the form of the covariance matrix. The program for composite likelihoods included in Appendix A is capable of performing these tasks.

5.2 The Moran/Coalescent Model

As with the Galton-Watson branching process approximations, Moran/Coalescent approximations require the first and second moments of the marker allele frequencies, along with the Hessian of Equation 3.21. The general form of the first two moments of the Moran model, derived via the coalescent, are found in Equations

4.16, 4.25 and 4.26. Finding the specific forms of these terms will make it possible to compute likelihoods for r, and derive point and interval estimates, provided additional assumptions are made. These assumptions are that the disease mutation occurred t generations in the past, and that the ancestral disease chromosome carried the most common marker allele in the actual disease population, or

$$\pi(0) \;=\; \mathbf{e}_1. \tag{5.25}$$

This model has an advantage over the branching process model. Its moments are in terms of the marker allele frequencies, rather than the marker allele counts. This makes it possible to use the likelihoods in Equations 3.18 and 3.19, without relying on further approximations. With these likelihoods, the Hessian is unchanged from what is listed in Equation 3.22. The disadvantage to this model with respect to the Galton-Watson model is that it is necessary to perform numerical integration to obtain second-order estimates.

5.2.1 Recombination Only

This section constructs approximations to the likelihood for r. The approximations depend on functions of the matrix shown in Equation 4.42.

First Order Approximation

The first order approximation relies entirely on the time-continuous transition matrix, $\mathbf{P}(t)$, defined in Equation 4.15. In order to find $\mathbf{P}(t)$, first note that

$$\begin{aligned}\mathbf{Q}^n &= r^n[P-\mathbf{I}]^n \\ &= r^n\left[(-1)^n\mathbf{I} + P\sum_{i=1}^{n-1}\frac{n!}{i!(n-i)!}(-1)^{n-i} + P\right] \\ &= r^n\left[(-1)^n\mathbf{I} + P\{-1-(-1)^n\} + P\right] \\ &= (-r)^n\left[\mathbf{I} - P\right], \qquad\qquad n = 1, 2, \ldots.\end{aligned} \tag{5.26}$$

From this, it is clear that

$$\begin{aligned}\mathbf{P}(t) \;=\; e^{\mathbf{Q}t} &= \mathbf{I} + \mathbf{Q}t + \tfrac{\mathbf{Q}^2t^2}{2!} + \tfrac{\mathbf{Q}^3t^3}{3!} + \ldots \\ &= P + [\mathbf{I}-P](-rt) + [\mathbf{I}-P]\tfrac{(-rt)^2}{2!} + [\mathbf{I}-P]\tfrac{(-rt)^3}{3!} + \ldots \\ &= P + [\mathbf{I}-P]e^{-rt}, \qquad\qquad t \geq 0.\end{aligned} \tag{5.27}$$

This provides the result that $\pi(t) = \mathbf{e}_1\{P + [\mathbf{I}-P]e^{-rt}\}$. Note that these are the same expected allele frequencies as obtained with the first-order approximation to the recombination-only model of Xiong and Guo [76] (see Equation 3.17). The first order approximation to the likelihood can now be written as

$$l(r) \;=\; n_{1d}\log[p_{1d} + (1-p_{1d})e^{-rt}] + \sum_{i=2}^{k} n_{id}\log[p_{id}(1-e^{-rt})]. \tag{5.28}$$

Differentiating the log-likelihood, and equating it to zero, results the expression

$$\frac{(n_d - n_{1d})e^{-\hat{r}t}}{1-e^{-\hat{r}t}} \;=\; \frac{n_{1d}(1-p_{1n})e^{-\hat{r}t}}{p_{1n} + (1-p_{1n})e^{-\hat{r}t}}, \tag{5.29}$$

where $n_d = \sum_{i=1}^{k} n_{id}$. Solving this for $\hat{r}$, produces a point estimate for the recombination fraction,

$$\hat{r} \;=\; -\frac{1}{t}\log\left(\frac{p_{1d}-p_{1n}}{1-p_{1n}}\right). \tag{5.30}$$

This estimate is very similar to the one obtained under the branching process model. It solves the formula

$$e^{-\hat{r}t} \;=\; \left(\frac{p_{1d}-p_{1n}}{1-p_{1n}}\right), \tag{5.31}$$

while the branching process estimate solves

$$(1-\hat{r})^{t} \;=\; \left(\frac{p_{1d}-p_{1n}}{1-p_{1n}}\right). \tag{5.32}$$

Since $\hat{r}$ is small, the two methods produce estimates that are almost equal. Just like the estimate from the Galton-Watson model, this first-order maximum-likelihood estimate is a simple function of a known measure of linkage disequilibrium.

Approximate confidence intervals are derived by taking the values of r whose log-likelihood values are two less than that of the maximum, in accordance with maximum-likelihood theory. A computer program that calculates the point and interval estimates for this result from the Moran model is included in Appendix A.

Second Order Approximation

It is possible to construct first-order approximations to the likelihood because a closed-form expression for $\mathbf{Q}_r^t$ exists(see Equations 5.26 and 5.27). However, because exponential growth of the disease population is assumed, it is impossible to form analytic expressions for the second-order approximation, as it requires solution of integrals of the form $\int e^{-e^{-x}} \mathrm{d}x$ (see Equation 4.22).

The second-order approximations are obtained by performing numerical integration for the elements of $\mathbf{R}(t)$, where $\mathbf{P}(t)$ is given in Equation 5.27. These results then produce the covariance structure of the allele frequencies, as listed in Equations 4.25 and 4.26. The computer program for composite likelihoods in Appendix A performs these tasks.

5.2.2 Recombination plus Mutations at Marker and Disease Loci

Referring to the applications of the Moran/Coalescent model in Chapter 4, it is possible generalize to allow for mutations at the marker and disease loci. The models generated by these assumptions are governed by the transition matrices, which are in turn controlled by the intensity matrices in Equations 4.43 and 4.45. This section mentions first- and second-order approximations to the likelihood for r using these models.

First Order Approximation

The possibility of general mutation processes at the marker loci, make it impossible to find closed-form expressions for $\mathbf{Q}^t_{mr}$ or $\mathbf{Q}^t_{mrd}$. Hence, analytic versions of the Markov transition matrices, $\mathbf{P}(t)$ are unavailable. Once again, the computer makes estimation possible. The program for composite likelihoods in Appendix A computes the correct version of $\mathbf{P}(t)$, and uses the result to form the first-order approximation.

Second Order Approximation

The second-order approximation relies not only on the form of $\mathbf{P}(t)$, but also on the form of $\mathbf{R}(t)$. This suffers from the additional complexity of requiring numerical integration. The computer program for composite likelihood estimation included in Appendix A performs the calculations to form the desired approximation.

Chapter 6

Mapping a Disease Gene with Multiple Markers

The previous chapters discussed estimating the distance between a disease gene and a single marker locus. However, data from several markers in the same region are often available (see it e.g. [39], [23], [51]). Mapping a disease gene by considering several markers simultaneously can potentially provide more information than relying on an aggregation of results obtained through single-marker estimation. For example, the consideration of several markers may make it possible to jointly estimate the location and the age of a disease mutation.

One can estimate the location of disease-influencing mutations with multiple markers with one of two types of data: single-marker or haplotype. An individual's alleles are typically identified one marker at at time, therefore single-marker data is more readily available than haplotype data. However, haplotype data is becoming increasingly available (see it e.g. [30], [72]). This chapter discusses methods for mapping disease genes with data from multiple markers.

6.1 Composite Likelihoods

In the absence of haplotype data in the region of interest, two options present themselves. The first option is to estimate the distance between the disease mutation and each marker individually. The initial search region is then defined as the intersection of all single-marker search regions. If the intersection is empty, one can start by searching the region around the marker that exhibits the most disequilibrium with the disease. The second option is to define a search region for the disease mutation using data from all of the markers simultaneously. This second approach is the more appealing of the two, but it requires additional machinery to combine the information from all of the markers.

Composite likelihood methods provides a mechanism to form search regions using the second option. As mentioned in Chapter 3, composite log likelihoods combine information from several, possibly dependent, sources by adding together conditional or marginal log likelihoods. Previous chapters developed marginal log likelihoods for single markers. The sum of these log likelihoods produces an instrument that estimates the location of disease genes through the combined information from several markers. What follows illustrates how to form a composite likelihood with data from a collection of markers.

Suppose that disease and normal chromosomes have been typed at K marker loci that are ordered on a chromosome as $l_{m1}, l_{m2}, \ldots, l_{mK}$, with distances in Morgans between markers $i-1$ and i given by m_i for $i = 2, 3, \ldots, K$. The Haldane map function translates these map distances into recombinatorial distances, $h(m_i)$ for $i = 2, 3, \ldots, K$ (see Equation 2.7). If x_d represents the map distance between marker locus l_{m1} and the disease locus, the recombinatorial distance is $h(x_d)$. Similarly, the recombination fraction between the disease gene and any marker other than the first is given by

$$r_i \;=\; \frac{1}{2}\left[1-\exp\left(-2\left|x_d-\sum_{j=2}^{i} m_i\right|\right)\right], \qquad i=2,3,\ldots,K. \tag{6.1}$$

If we let L_i denote the marginal likelihood function of r_i, obtained from any of the models in Chapter 5, then the composite log likelihood is defined as

$$cl \;=\; \sum_{i=1}^{K} \log(L_i). \tag{6.2}$$

Using the recombination fractions r_i from Equation 6.1 in the marginal log likelihoods, and combining the log likelihoods as in Equation 6.2, produces the value of the composite log likelihood for any hypothetical position of the disease locus. Repeating this procedure on a grid of points along the genetic map produces a numerical representation of the composite likelihood. Evaluation of the composite likelihood along a grid provides several benefits. The most apparent is the numerical ease of computation. It also allows the researcher to consider local, as well as global, maxima when determining a region to begin the physical search for disease genes.

Because composite log likelihoods share some of the properties of ordinary log likelihoods they can be used to obtain approximate confidence regions. An approximate confidence interval is therefore the region where the composite log likelihood is greater than the maximal value of the composite log likelihood minus two, just as with traditional likelihoods. However, at times it is instructive to form more conservative confidence bounds. One very conservative approach is to define the search area as the region where the composite log likelihood is greater than the maximal value of the composite log likelihood minus two times the number of markers.

As an illustration, consider a subset of the data collected in the search of a gene for Diastrophic Dysplasia [24] (Chapter 7 presents estimates based on all of the data). In this example, a composite likelihood is used to define a refined search area for the disease gene in the region bounded by D5S372 and CSF1R/CCT. This region spans 0.00866 Morgans, or about 866 kb. Applying Equation 6.2 to the data in Table 6.1 along a grid of possible locations of the disease gene,

Marker	Morgans to	Disease	Normal
	Next Marker	Counts	Counts
D5S372	0.00775	93	16
		61	103
BT1	0.00045	139	5
		13	117
CSF1R/EcoRI	0.00046	150	12
		8	116
CSF1R/CCT	-	97	5
		27	125

Table 6.1 Data from a selection of markers found to be associated with presence of Diastrophic Dysplasia, taken from Hästbacka, et al. (1992).

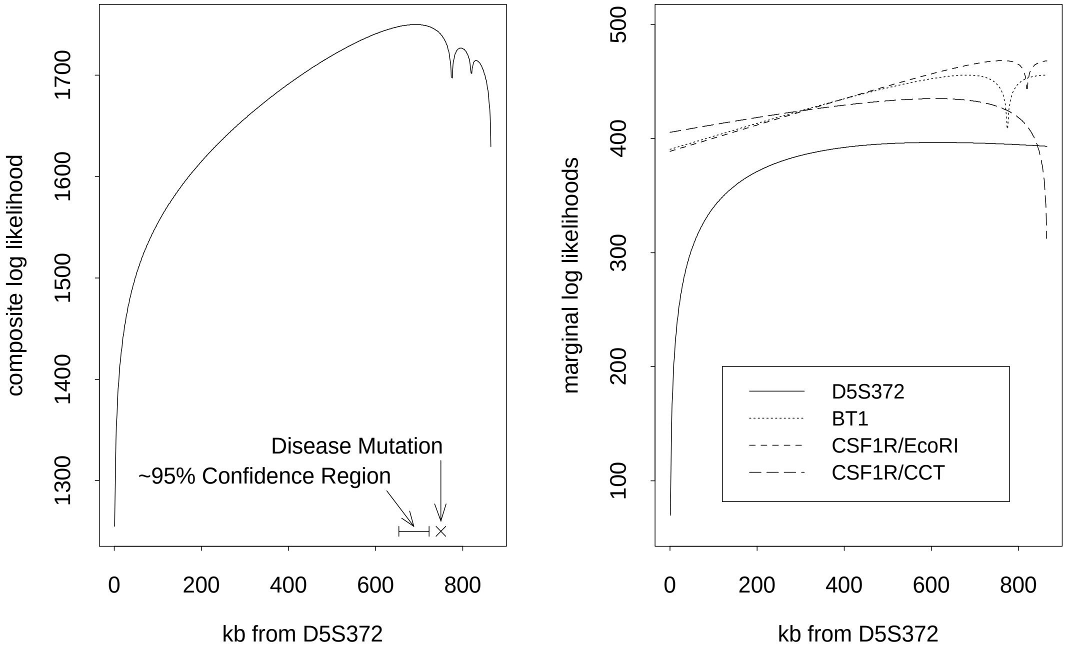

Figure 6.1 Example composite likelihood for four markers associated with Diastrophic Dysplasia, including the marginal log likelihoods.

and assuming that the disease mutation occurred 100 generations in the past, produces the composite log likelihood. The first graph in Figure 6.1 displays the composite likelihood, based on the first-order approximation to the Galton-Watson recombination-only model, and computed along a grid with a mesh size of 1 kb. The second shows the contribution from each marginal log likelihood. Note that all of the component marginal log likelihoods have relatively high values in the region where the composite log likelihood achieves its maximum.

The initial search region, as defined by an approximate 95% confidence interval, occupies a stretch of DNA beginning about 653 kb from the D5S372 locus, and ending about 723 kb from it. The disease gene was located by Hästbacka et al. [23] at a distance of about 750 kb from the D5S372 marker locus. The disease gene was outside the confidence region, but only by about 30 kb.

6.1.1 Smoothing the Composite Likelihood

The composite log likelihood in Figure 6.1 demonstrates a complication that arises when summing likelihoods of the types derived in Chapter 5. Those likelihoods are constructed assuming that the marker locus is not the disease locus and that the location of the marker is precisely known. The difficulty that arises from these assumptions is that each marginal log likelihood must go to negative infinity at the position of the marker locus. The composite log likelihood therefore can have very small values near marker loci, especially if several markers are close together.

To illustrate this, assume that the disease is very near to the marker locus l_{mi}. Even if all of other the marginal log likelihoods indicate that the region is likely to contain the disease mutation, the composite log likelihood may not be maximized in that region, due to the extremely small values of the likelihood near l_{mi}.

One way to overcome this is to smooth the composite log likelihood. This smoothing can be achieved in a variety of ways, some of them statistical and some more intuitive or ad hoc.

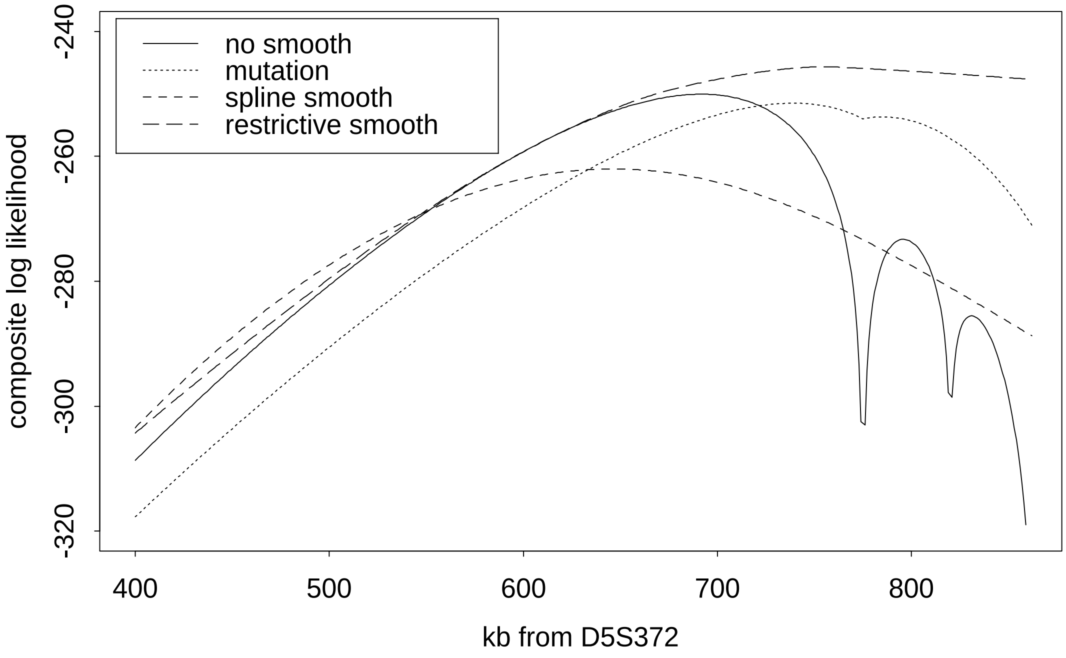

Figure 6.2 A range-restricted view of the composite likelihood, both smoothed and unsmoothed, from Figure 6.1.

The graph in Figure 6.2 depicts a region of the composite likelihood in Figure 6.1. In addition to showing the composite likelihood, it gives the results from three possible "smoothing" schemes. The mutation "smooth" was obtained by setting the mutation matrix for each of the marker loci equal to

$$\begin{pmatrix} 0.9995 & 0.0005 \\ 0.0005 & 0.9995 \end{pmatrix}. \tag{6.3}$$

The spline smooth is representative of other classical smoothing techniques, where one must over-smooth considerably to diminish the effects of the small values near the marker. The restrictive smooth reflects an intuitive way to smooth the marginal log likelihoods. The reasoning behind it is as follows.

1. Consider a marker that is extremely close to the disease mutation.
2. The proximity of the marker to the disease locus will precipitate small log likelihood values near the true location of the disease mutation.
3. In order to eliminate the small values near the marker, set all values of the log likelihood between the marker locus and the maximizer that are less than some specified value equal to that value.

In the examples given in Figures 6.2 and 6.5, all values of the log likelihood between the marker locus and the maximizer are set equal to the maximum value. With this smoothing technique, it becomes difficult to obtain lower limits to the search relative to a single marker. However, the consequences arising from this are diminished when more than one marker is used.

It is interesting to note that allowing for the occurence of marker mutations results in a smoothed composite log likelihood. The explanation for this is that mutations are highly confounded with recombination events. In other words, it is difficult to distinguish between mutation and recombination events in models for a single marker locus. Therefore, if mutations are possible, then fewer recombinations are needed to explain the distribution of allele frequencies. This causes the component log likelihoods to peak nearer to the marker loci, diminishing the problem of hugely negative values near marker loci.

Smoothing the composite log likelihood is appropriate in the current example, where the disease mutation is about 25 kb from the BT1 marker locus. However, smoothing produced mixed results. The spline smooth produced an estimate that was even farther away from the marker than the unsmoothed version (104, rather than 59 kb away). The mutation and restrictive smooths provided good improvements, missing the truth by only 12 and 5 kb respectively. It would seem that smoothing, if done appropriately, can improve estimates. The obvious choice based on the example presented here is to smooth by imposing a mutation mechanism on the marker loci. However, one successful attempt is not enough to establish the general usefulness of this technique. Chapter 7 further examines the the effect of smoothing by applying the mutation and restrictive smoothing techniques to other data sets.

6.1.2 Joint Estimation of the Age and Location of a Disease Mutation

Since the composite likelihood utilizes data from more than one marker locus, it may be possible to estimate the age of the disease mutation jointly with its location. This can be done quite simply with methods previously mentioned. In the work already presented, the age is assumed to be a constant. In order to estimate the age of a disease mutation jointly with its location, all that must be done is to compute the composite log likelihood along a grid of disease ages in tandem with the grid of disease gene locations. Computing the value of the composite log likelihood at the points of the two-dimensional grid produces a likelihood surface that jointly estimates the age and the location of a disease mutation.

To illustrate this, consider again the data in Table 6.1. Rather than fixing the age parameter at 100 generations, the calculations were carried out along a grid of values, ranging from 10 to 300 in steps of 10 generations. Figures 6.3 through 6.6 contain contour plots of the joint composite log likelihood for age and location, produced by different smoothing mechanisms. The first contour in each plot defines an approximate ninety-five percent confidence region for the parameters, except in the restrictive smooth plot, where the second contour defines the confidence region. The point identified as truth on the plots is approximate for both the location and

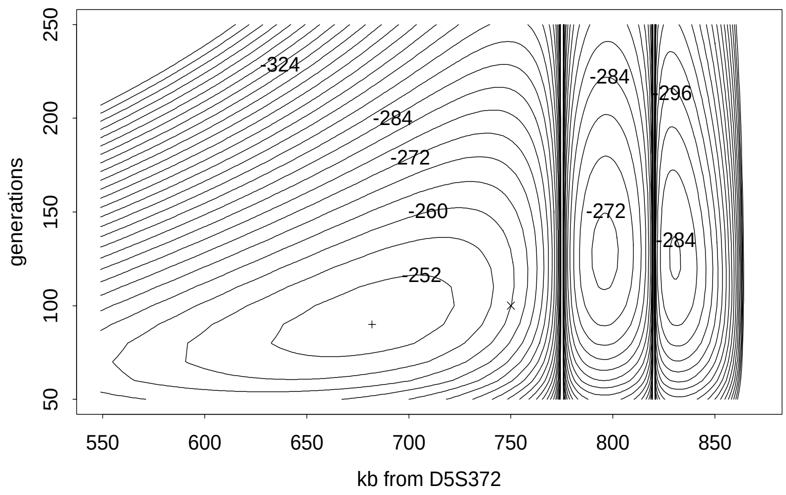

Figure 6.3 Contour plot of a joint composite log likelihood for location and age. The maximum likelihood estimate is marked with +, and the truth is marked with ×.

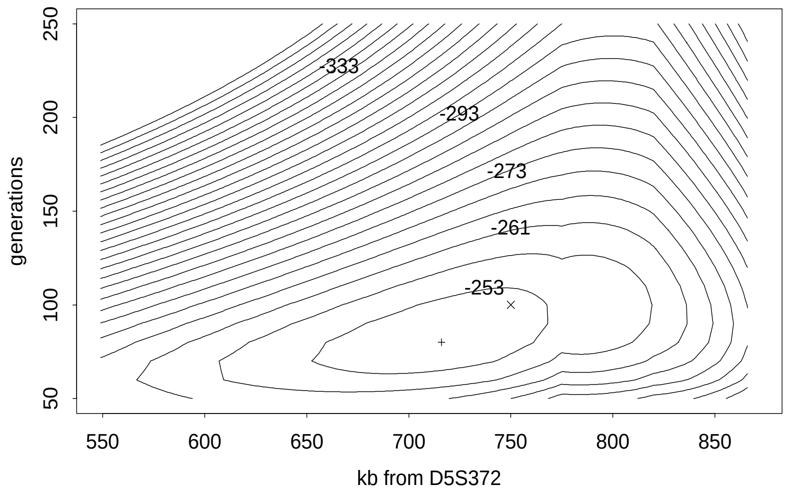

Figure 6.4 Contour plot of a smoothed version (mutation smooth) of the joint composite log likelihood from Figure 6.3. The maximum likelihood estimate is marked with +, and the truth is marked with ×.

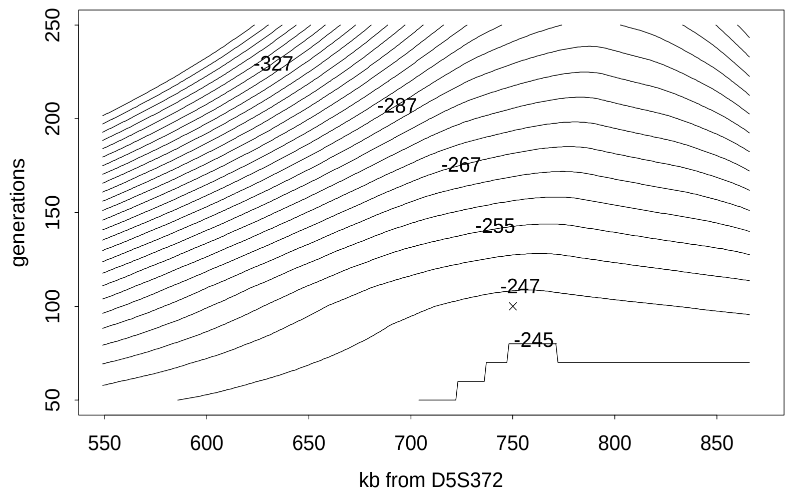

Figure 6.5 Contour plot of a smoothed version (restrictive smooth) of the joint composite log likelihood from Figure 6.3. The contour marked by straight lines denotes a flat region where the joint composite log likelihood is maximized. The truth is marked with ×.

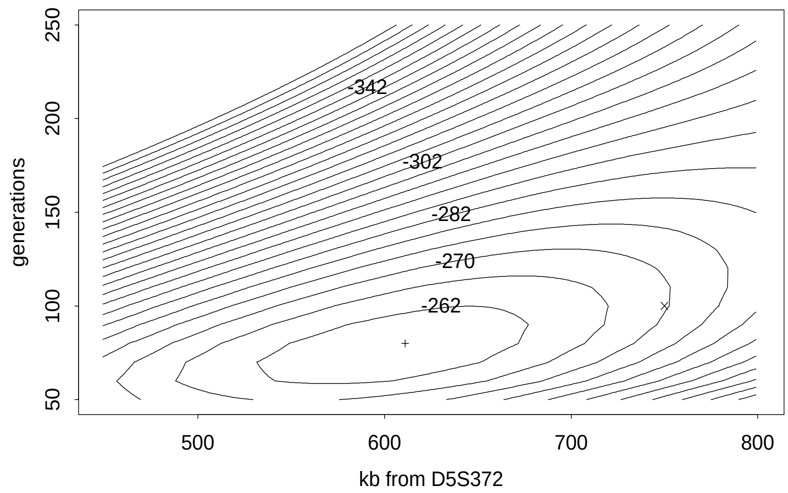

Figure 6.6 Contour plot of the joint composite log likelihood from Figure 6.3, smoothed with beta splines. The maximum likelihood estimate is marked with +, and the truth is marked with ×.

the age of the disease mutation. First, chromosomes of different individuals are not exactly identical. Also, the "true" age of the disease is itself only an estimate obtained through other means [24].

This brief study shows that it might be possible to jointly estimate the age and location of a disease mutation through use of joint composite log likelihoods. However, when the maximal restrictive smooth produces contours that are not closed, making it impossible to estimate the age of the ancestral disease mutation. This is probably due to an over-smoothing effect that blurs the locations of the marker loci, making it difficult to use recombinations between them to count the generations that have passed.

The results from the other joint log likelihoods are varied. They produce 95% confidence contours that are closed, but only when the surface is smoothed via mutation does the contour contain the truth. This reinforces the results obtained via smoothing when the age was fixed. It suggests that including mutation processes at the marker loci is the preferred course to take when using composite likelihood methods.

The usefulness of composite log likelihoods will be examined later in Chapter 7 by applying them to data. This will shed some light on the capability of this methodology to correctly identify both the age and location of a disease-influencing mutation.

6.2 Haplotype Models

If true haplotype data is available, then composite likelihoods do not utilize all of the possible information found in the data. Hence, estimation techniques that are geared directly to haplotype data should provide better results than are possible with single-marker and composite likelihood methods.

However, it can be difficult to form estimation procedures based on haplotype data using the methodology presented in this thesis for single markers. This is because the moment and intensity matrices require that the transition probabilities from one haplotype to another be specified. For example, if the map consists of nine marker loci, each having two alleles, it is necessary to account for 512 haplotypes. Further, a specific transition law must be obtained for each possible location of the disease mutation. For example, if the disease mutation lies between markers 3 and 4, the transition probabilities differ from those induced when the mutation occurred between markers 4 and 5. The next section clarifies these problems by considering the two-marker haplotype transition probabilities.

6.2.1 Two-Marker Transitions

This section describes how to obtain haplotype transition probabilities in the situation when mutation effects are absent. The list below enumerates several required assumptions.

1. A disease-influencing mutation is located in a region near two markers, A, possessing a alleles, and B, with b alleles.

2. The linked markers are ordered on chromosomes as A-B.

3. The population of normal chromosomes is larger than the subpopulation carrying the disease mutation.

4. The A-B haplotype frequencies are fixed quantities in the general population.

5. At some time in the past, a single disease chromosome existed in the entire population.

6. The ancestral disease chromosome carried the haplotype that is the most common haplotype within the current disease population.

7. Both markers in the haplotype are linked to the disease mutation. The recombination rate between the disease gene and marker A is r_A, the recombination rate between the disease gene and marker B is r_B and the recombination rate between the two marker loci is r_{AB}.

8. The subpopulation of disease chromosomes is growing exponentially, at a rate of $1 + \lambda$, where $\lambda > 1$.

Applying these assumptions in tandem with the assumption of random mating, either in a Wright-Fisher or Moran sense, produces the transition matrices. Since the transition probabilities depend on the location of the disease gene relative to the marker loci, one must consider three possible two marker-plus-disease haplotypes. These are D-A-B, A-D-B and A-B-D, where D represents the disease gene. Let's assume that if there are a alleles at marker A and b alleles at marker B, then the haplotypes are ordered as $\mathrm{A}_1\mathrm{B}_1$, $\mathrm{A}_1\mathrm{B}_2$, …, $\mathrm{A}_1\mathrm{B}_b$, $\mathrm{A}_2\mathrm{B}_1$, …, $\mathrm{A}_a\mathrm{B}_b$.

The following matrices will be needed to specify the transition matrices. They are analogous to the matrix expressed in Equation 4.28.

$$P_A = \begin{pmatrix} p_{A_1 n} & p_{A_2 n} & \cdots & p_{A_a n} \\ p_{A_1 n} & p_{A_2 n} & \cdots & p_{A_a n} \\ \vdots & \vdots & & \vdots \\ p_{A_1 n} & p_{A_2 n} & \cdots & p_{A_a n} \end{pmatrix}, \tag{6.4}$$

$$P_B = \begin{pmatrix} p_{B_1 n} & p_{B_2 n} & \cdots & p_{B_a n} \\ p_{B_1 n} & p_{B_2 n} & \cdots & p_{B_a n} \\ \vdots & \vdots & & \vdots \\ p_{B_1 n} & p_{B_2 n} & \cdots & p_{B_a n} \end{pmatrix} \tag{6.5}$$

and

$$P_{AB} = \begin{pmatrix} p_{A_1 B_1 n} & p_{A_1 B_2 n} & \cdots & p_{A_1 B_b n} & p_{A_2 B_1 n} & \cdots & p_{A_a B_b n} \\ p_{A_1 B_1 n} & p_{A_1 B_2 n} & \cdots & p_{A_1 B_b n} & p_{A_2 B_1 n} & \cdots & p_{A_a B_b n} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ p_{A_1 B_1 n} & p_{A_1 B_2 n} & \cdots & p_{A_1 B_b n} & p_{A_2 B_1 n} & \cdots & p_{A_a B_b n} \end{pmatrix}. \tag{6.6}$$

In these equations, $p_{A_i n}$ are the normal allele counts for marker A, $p_{B_j n}$ are the normal allele counts for marker B and $p_{A_i B_j n}$ are the haplotype counts for the normal population.

Considering the order D-A-B, and allowing only recombination events to change the type of the disease chromosomes, produces the simplest version of the transition matrix. Consider the haplotype A$_i$-B$_j$. The probabilities for the A$_i$-B$_j$ $\rightarrow$ A$_i$-B$_j$ and A$_i$-B$_j$ $\rightarrow$ A$_k$-B$_l$ transitions are

$$\begin{aligned} p_{i\rightarrow i, j\rightarrow j} &= (1-r_A)(1-r_{AB}) + (1-r_A)r_{AB}p_{B_j n} + r_A(1-r_{AB})p_{A_i B_j n} \\ &\quad + r_A r_{AB} p_{A_i n} p_{B_j n}, \qquad i=1,\ldots,a; j=1,\ldots,b, \end{aligned} \tag{6.7}$$

and

$$\begin{aligned} p_{i\rightarrow k, j\rightarrow l} &= (1-r_A)r_{AB}p_{B_l n} + r_A(1-r_{AB})p_{A_k B_l n} \\ &\quad + r_A r_{AB} p_{A_k n} p_{B_l n}, \quad i=1,\ldots,a; j=1,\ldots,b; (k\neq i)\cup(l\neq j), \end{aligned} \tag{6.8}$$

respectively. Combining these expressions to form a matrix, produces the compact expression of the transition law,

$$\begin{aligned} \mathbf{T}_{DAB} &= (1-r_A)(1-r_{AB})\mathbf{I}_{ab} + (1-r_A)r_{AB}\left\{\mathbf{J}_a \otimes P_B\right\} \\ &\quad + r_A(1-r_{AB})P_{AB} + r_A r_{AB}\left\{P_A \otimes P_B\right\}, \end{aligned} \tag{6.9}$$

where $\mathbf{I}_{ab}$ is an identity matrix of dimension $ab \times ab$, $\mathbf{J}_a$ is a matrix of ones with dimension $a \times a$ and $\otimes$ represents the Kronecker product. Recall that the Kronecker product of two matrices, **A** and **B**, is defined as

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} A_{11}\mathbf{B} & \dots & A_{1a}\mathbf{B} \\ \vdots & & \vdots \\ A_{a1}\mathbf{B} & \dots & A_{aa}\mathbf{B} \end{pmatrix}. \tag{6.10}$$

The transition matrices for the other orders are obtained similarly. In particular, if the haplotype has the order A-B-D, the transition matrix is

$$\begin{aligned} \mathbf{T}_{ABD} &= (1-r_{AB})(1-r_B)\mathbf{I}_{ab} + (1-r_{AB})r_B P_{AB} \\ &\quad + r_{AB}(1-r_B)\left\{P_A \otimes \mathbf{J}_B\right\} + r_{AB}r_B\left\{P_A \otimes P_B\right\}. \end{aligned} \tag{6.11}$$

Likewise, the transition matrix for haplotypes of order A-D-B is

$$\mathbf{T}_{ADB} = \left[(1-r_A)\mathbf{I}_a + r_B P_A\right] \otimes \left[(1-r_B)\mathbf{I}_b + r_B P_B\right]. \tag{6.12}$$

With these matrices, one can obtain moment and/or intensity matrices as in Chapter 4 for single marker transitions.

6.2.2 Multiple Marker Transitions

If haplotype data for a genetic map with more than two markers is available, one can attempt to construct general matrices of haplotype transition probabilities. However, the work in the previous section hints at several difficulties. First, if the haplotype consists of K markers and the i^{th} locus has k_i alleles, then the transition matrix is of dimension $\prod_{i=1}^{K} k_i \times \prod_{i=1}^{K} k_i$, which can be very large. Second, a separate transition matrix is required for each of the $K+1$ possible locations of the disease gene relative to the K marker loci. Third, each transition matrix will need to be constructed through Kronecker products up to the K^{th} order.

Single-marker data is more readily available than haplotype data and composite likelihoods are more tractable than haplotype likelihoods. Therefore, the preferred approach for estimating the location of disease genes is to use composite likelihoods (see *e.g.* [12] and [76]).

6.3 Multiple Marker Simulations

While it can be very difficult to enumerate explicit transition and moment matrices for use in the procedures presented in this thesis for single matrices, it is relatively easy to define a simulation algorithm that captures the features of those transition matrices. As it will be necessary to simulate haplotype data to evaluate composite log likelihoods, this section defines a population model based on a Wright-Fisher sampling scheme.

6.3.1 Population Assumptions

The assumptions required for this model are somewhat different from what those of other population models previously described. They are outlined below.

1. A disease-influencing gene is located near, or within, a region of a chromosome spanned by genetic markers $l_1, l_2, \ldots, l_K$, which carry $k_1, k_2, \ldots, k_K$ alleles, respectively.

2. The order of markers is known, and specified in (1). The distance in Morgans between markers $i-1$ and i are given by m_i for $i = 2, 3, \ldots, K$.

3. The population of normal chromosomes is larger than the subpopulation carrying the disease mutation.

4. The haplotype frequencies are fixed quantities in the general population.

5. At some time in the past, a single disease chromosome existed in the entire population.

6. The subpopulation of disease chromosomes grows deterministically in discrete time at a rate of $1 + \lambda$ per generation, with $\lambda > 0$.

7. Mutations may occur at any locus in the map. This map includes the marker loci and the disease gene.

8. The mutation rates and patterns are known, with $\mathbf{U}_i$ representing the mutation matrix of marker l_i for $i = 1, 2, \ldots, K$ and ν_d representing the rate of new disease mutations.

9. During reproduction, recombinations occur within the haplotype according to a homogeneous Poisson process, with intensity equal to one recombination event per Morgan per meiosis. This is the assumption that produces the Haldane mapping function.

10. Production of subsequent generations occurs according to a Wright-Fisher sampling scheme.

These general assumptions, make it possible to write a haplotype simulation algorithm.

6.3.2 Simulation Algorithm

This section sketches the algorithm to be used for simulating the expansion of disease chromosomes within a population, based on the assumptions listed in the previous section.

1. Fix all the parameters of the model. This includes the size of the disease population, the age of the disease, the number of markers and their intermarker distances, the location of the disease mutation and the mutation rates and patterns for each of the loci in question.

2. Repeat steps (a) and (b) below for each of the generations of the age of the disease ($t = 1, \ldots,$ age).

    (a) Calculate the number of disease chromosomes in the disease population as $X_T(t) = X_T(0)(1+\lambda)^t$.

    (b) Repeat steps i through vi below for each of the $X_T(t)$ disease chromosomes that will comprise the t^{th} generation.

i. Randomly, and with replacement, select a haplotype out of those in the prior generation. It will produce a single offspring haplotype.

ii. Mutate each of the markers independently, according to its mutation rate and pattern.

iii. Generate locations of recombination within the haplotype according to a homogeneous Poisson process, as in Assumption 10 in the previous section.

iv. Retain the segment of the haplotype that contains the disease gene.

v. If needed, add new fragments of DNA to complete the new disease haplotype. The sequences of alleles that are added on correspond to the frequencies of those fragments in the population of normal haplotypes.

vi. Add new disease mutations by inserting a randomly selected normal haplotype into the disease population at the rate of ν_d per meiosis.

This algorithm makes it possible to directly simulate the expansion of disease haplotypes in a population. It will be used to simulate samples from a population of disease chromosomes. Those samples will make it possible to evaluate composite likelihood methodology in the next chapter.

Chapter 7

Application to Data

This chapter studies the techniques described in this thesis by applying them to a variety of data sets, both simulated and real.

7.1 Simulated Data

This section examines the behavior of the estimators from this thesis with regards to simulated populations. The first subsection studies the single marker case. It first uses samples generated after simulating a population of disease chromosomes with the algorithm of Kaplan *et al.* [36]. The primary purpose of this exercise is to compare the closed-form first order approximation to the simulation estimates proposed by Kaplan, Hill and Weir [36] when the true model is a branching process. The subsection also obtains samples from data simulated from the haplotype model described in the previous chapter. With these samples, it compares the estimating procedures for single-marker data described in this thesis. The second subsection considers multiple linked markers. It uses samples obtained from populations simulated using the haplotype algorithm described in Chapter 6 to evaluate the composite likelihoods discussed previously.

7.1.1 Single Marker

Branching Process: Simulation vs. First Order Approximation

The simulation technique proposed by Kaplan *et al.* [36] estimates the recombination fraction based on the entire history of a branching process. The first order approximation uses mean information, making it possible to form analytic estimators when there is no mutation. This section compares the behavior of the point estimates derived from these two different techniques.

This comparison assumes that the expansion of disease chromosomes within the total population follows a Galton-Watson branching process. It also assumes that no mutations occur after the ancestral disease chromosome is introduced into the population. The section presents the results from simulations for a number of fixed recombination fractions. The results were obtained by simulating the expansion of the disease population and estimating the recombination rate from a sample collected from the simulated disease population.

Table 7.1 contains the means and the square-roots of the mean squared errors for the estimates from fifty simulations at each of eight selected recombination fractions. The numbers in the table lead to several observations. The first is that the point estimates obtained from the first-order approximation appear to be slightly smaller than the truth. The second is that the simulated point estimates exhibit an increasing upward bias as the recombination fraction increases. This

Truth	Simulation		First-Order	
	mean	rtMSE	mean	rtMSE
0.0005	0.000494	0.00020	0.000402	0.00031
0.0010	0.001159	0.00135	0.000816	0.00060
0.0015	0.001448	0.00049	0.001174	0.00064
0.0020	0.002534	0.00365	0.001667	0.00119
0.0025	0.002937	0.00333	0.002098	0.00202
0.0050	0.007481	0.00935	0.005034	0.00279
0.0075	0.009326	0.01002	0.006752	0.00352
0.0100	0.017169	0.02258	0.009509	0.00456

Table 7.1 Means and root mean squared errors for simulation and first-order approximation estimates from 50 simulations at each of 8 selected recombination coefficients.

leads to the final observation: the bias in the simulated estimates causes the mean-squared error of the simulator estimates to be significantly greater than the mean-squared error of the first-order approximation estimates for larger recombination rates.

Summing the mean-squared errors across the various recombination coefficients produces an "integrated" root-mean-squared error of 0.00688 for the approximation estimator. This value is smaller than the value of 0.0148 produced by the simulation estimator, indicating that the first-order estimator outperforms the simulation estimator of Kaplan *et al.* [36].

Branching Process and Moran/Coalescent Approximations

The last section compared the first order branching process approximation to a simulation-based estimation procedure. This section uses simulated data to compare the four approximation estimators of this thesis to each other.

Table 7.2 contains the parameters used to perform a simulation according to

Parameter	value
sample size	150
alleles at each locus	2
mutation rate at each locus	0
disease mutation rate	0
kb from first marker to disease gene	275
age of disease	200

Table 7.2 Parameters set in simulation for single-marker comparisons.

the haplotype model described in Chapter 6 for a map of ten markers spanning a region of 500 kilobases. Estimating the location of the disease mutation for each of the markers separately produced the results contained in Tables 7.3 and 7.4.

As seen in Tables 7.3 and 7.4, three of the estimation techniques produce results that are almost identical: the Galton-Watson and Moran/Coalescent first-order approximations (FOA) and the Moran/Coalescent second-order approximation (SOA). Only the Galton-Watson SOA differs significantly from the others.

Locus	Truth	Galton-Watson		Moran/Coalescent	
		FOA	SOA	FOA	SOA
1	275	302	1700	303	304
2	160	183	1195	183	184
3	45	72	260	72	72
4	35	79	285	79	80
5	25	14	90	14	14
6	15	42	175	42	42
7	5	7	60	7	7
8	5	14	90	14	14
9	115	146	620	146	147
10	225	411	>2000	412	413

Table 7.3 Comparison of point estimates from the four approximation techniques for a simulated map of ten markers. All distances are given in kb. Recall that FOA represents a first-order approximation and that SOA represents a second-order approximation. The grid for the Galton-Watson second order approximations was set at 5 kb to speed computations.

Locus	Truth	Galton-Watson		Moran/Coalescent	
		FOA	SOA	FOA	SOA
1	275	(198,458)	(1040,2000+)	(198,459)	(199,460)
2	160	(112,287)	(610,2000+)	(112,288)	(113,289)
3	45	(34,131)	(195,365)	(34,131)	(35,132)
4	35	(39,142)	(210,405)	(39,142)	(40,143)
5	25	(2,44)	(60,130)	(2,44)	(2,44)
6	15	(16,88)	(130,245)	(16,88)	(16,89)
7	5	(0,31)	(35,90)	(0,31)	(0,32)
8	5	(2,44)	(60,130)	(2,44)	(2,44)
9	115	(86,236)	(385,1775)	(86,236)	(86,237)
10	225	(274,628)	(1215,2000+)	(274,630)	(275,631)

Table 7.4 Comparison of interval estimates from the four approximation techniques for a simulated map of ten markers. All distances are given in kb. Recall that FOA represents a first-order approximation and that SOA represents a second-order approximation. The grid for the Galton-Watson second order approximations was set at 5 kb to speed computations.

The fact that the first-order approximations are similar is not surprising. Their concordance can be explained by the likelihoods themselves (see Equations 5.7 and 5.28), which differ only slightly since $e^{-rt} \approx (1-r)^t$.

The similarity between the first-order (FOA) and second-order (SOA) approximations derived from the Moran/Coalescent model is somewhat more surprising. However, this outcome is supported by the results of Xiong and Guo [76], whose first-order and second-order approximations are also similar.

The most notable difference is that the second order approximation based on the branching process model appears to be unstable. It produces estimates that are almost always far too large. The reason for this seems to be that there is a high degree of variability in the allele counts of the linked marker, even when the recombination rate is very low. This, coupled with the form of the Hessian (see Equation 3.22), produces very negative values of the log likelihood in regions neighboring each marker locus.

Estimates with Marker Mutations

Since the models of this thesis make it possible to allow for mutations at the loci, it is of interest to examine how the estimators perform when mutations are allowed at the marker loci. In this section, the matrix containing the transition probabilities from one allele (rows) to another (columns) is set equal to

$$\begin{pmatrix} 0.9995 & 0.0005 \\ 0.0005 & 0.9995 \end{pmatrix} \tag{7.1}$$

for each marker. This model that allows for mutations at the marker loci is then used to obtain estimates corresponding to those in Tables 7.3 and 7.4. The results are listed in Tables 7.5 and 7.6.

The effect of allowing marker mutations is clear: it reduces the estimate of the genetic distance. In the case where all of the mutation rates were assumed to be the same (0.0005), the estimated distances were decreased by about 100 kb. This is an indication of how highly confounded recombination and mutation are. Therefore, knowing mutation rates and patterns should improve genetic mapping efforts.

Locus	Truth	Galton-Watson		Moran/Coalescent	
		FOA	SOA	FOA	SOA
1	275	202	1610	203	203
2	160	83	1095	83	84
3	45	1	160	1	1
4	35	1	185	1	1
5	25	1	5	1	1
6	15	1	75	1	1
7	5	1	5	1	1
8	5	1	5	1	1
9	115	46	620	46	47
10	225	311	1715	312	313

Table 7.5 Comparison of point estimates, with mutation, from the four approximation techniques for a simulated map of ten markers. All distances are given in kb. Recall that FOA represents a first-order approximation and that SOA represents a second-order approximation. The grid for the Galton-Watson second order approximations was set at 5 kb to speed computations.

Locus	Truth	Galton-Watson		Moran/Coalescent	
		FOA	SOA	FOA	SOA
1	275	(78,358)	(805,>2000)	(98,359)	(98,360)
2	160	(12,187)	(510,>2000)	(12,187)	(13,188)
3	45	(0,42)	(95,265)	(0,42)	(0,42)
4	35	(0,48)	(110,300)	(0,48)	(0,49)
5	25	(0,19)	(0,30)	(0,19)	(0,19)
6	15	(0,26)	(30,140)	(0,26)	(0,26)
7	5	(0,18)	(0,20)	(0,18)	(0,18)
8	5	(0,19)	(0,30)	(0,19)	(0,19)
9	115	(0,135)	(285,1680)	(0,136)	(0,136)
10	225	(174,527)	(865,>2000)	(174,529)	(175,530)

Table 7.6 Comparison of interval estimates, with mutation, from the four approximation techniques for a simulated map of ten markers. All distances are given in kb. Recall that FOA represents a first-order approximation and that SOA represents a second-order approximation. The grid for the Galton-Watson second order approximations was set at 5 kb to speed computations.

The differences among the estimates are minimal, both with and without mutation, except for the second order branching process estimates. Therefore, the evaluations that remain use only the first order approximation to the branching process, unless otherwise stated.

7.1.2 Multiple Markers

The aim of this section is to examine the ability of the estimation methods of this thesis to locate disease genes when a map with more than one marker is available. It studies the case where there are two marker loci and then provides some illustrations where there are more than two markers.

Two-Marker Results

This section describes the results of a single replicate of a factorial experiment designed to examine the impact of various population parameters on composite likelihood estimates based on two marker loci. Table 7.7 contains the parameters, with the values used in the study.

The algorithm for haplotype simulation described in Chapter 6 was used to produce a population of disease chromosomes according to combinations of the parameters in Table 7.7. Two other parameters were fixed for all realizations of the experiment. The population of disease chromosomes was assumed to grow from one to 100000 at a deterministic rate. This number was selected for compu-

Parameter	Values examined
sample size (dsamp)	25, 150, 250
kb between markers (kb_b)	20, 200, 2000
alleles at locus 1 ($a1$)	2, 6
alleles at locus 2 ($a2$)	2, 8
mutation rate at locus 1 (m1)	0, 1×10^{-6}
mutation rate at locus 2 (m2)	0, 1×10^{-4}
disease mutation rate (md)	0, 1×10^{-7}
location of disease gene (kb_d)	-50, 10, 100, 1000
age of disease (age)	100, 200, 400

Table 7.7 Values of parameters examined in the factorial experiment for two-marker composite likelihoods.

tational convenience. This choice is not without precedent. It has been used by other researchers in population simulations (see *e.g.* [12] and [11]). Another fixed parameter was the marker allele frequencies in the total population. They were set equal to one over the number of marker alleles (i.e. if there were two alleles, then they each were assumed to have a frequency of 0.5). Another important assumption was the pattern of mutations at the marker loci. When mutations occurred at a marker locus, a single-step model of mutation with reflective boundaries was assumed. The specific form of the mutation matrix, conditional on a mutation occurring was

$$\begin{pmatrix} 0.4 & 0.6 & 0.0 & \dots & & 0.0 \\ 0.4 & 0.0 & 0.6 & 0.0 & \dots & 0.0 \\ & & \ddots & \ddots & \ddots & \\ 0.0 & \dots & 0.0 & 0.4 & 0.0 & 0.6 \\ 0.0 & \dots & & 0.0 & 0.4 & 0.6 \end{pmatrix}. \tag{7.2}$$

Once the samples from the simulated populations were obtained, the data were analyzed with three different composite likelihoods. The first estimates were from a composite likelihood calculated under the assumption of no mutations. The second set of estimates was computed by assuming that mutations occurred at the marker loci with a mutation transition matrix of

$$\begin{pmatrix} 0.9997 & 0.0003 & 0.0 & \dots & & 0.0 \\ 0.0002 & 0.9995 & 0.0003 & 0.0 & \dots & 0.0 \\ & & \ddots & \ddots & \ddots & \\ 0.0 & \dots & 0.0 & 0.0002 & 0.9995 & 0.0003 \\ 0.0 & \dots & & 0.0 & 0.0002 & 0.9998 \end{pmatrix}. \tag{7.3}$$

The final estimates were made using the maximal restrictive smooth described in Chapter 6.

Two behaviors of the estimates were of interest: the point estimates and the confidence intervals. To evaluate the quality of the point estimates, the distance between the predicted location and the truth, plus one, was logarithmically transformed to stabilize the variance and to achieve approximate normality. Even after performing this transformation, the fit of the models was insufficient. Fitting separate analysis of variance models to the data for each level of the disease location factor (kb_d), rectified this problem.

In order to analyze the data, it was necessary to pool high order interactions to form the error term. Using a forward selection procedure provided the general form of the models. Performing full and reduced model tests indicated that adding four-way interactions to the three-way interaction model for the kb_d=-50 factor level did not significantly improve the fit of the model. Likewise, the models including three-way interactions were not significantly better than the models with only two-way interactions for each of the other three levels of the kb_d factor. Therefore, for the kb_d=-50 factor level, a model including one-way, two-way and three-way interactions was selected. For the other three levels, models including one-way and two-way interactions were chosen.

The tables in Appendix C contain multivariate and univariate analysis of variance tables for each of the levels of kb_d and for each of the repeated measurements: the estimates based on no smoothing, mutation smoothing and restrictive smoothing. The main effects and interactions that were significant at a level of $\alpha = 0.01$ level in a multivariate analysis were considered to be important. The discussion that follows focuses on specific levels of the kb_d factor and considers only those terms that were significant in the multivariate model and all three of the univariate models for the repeated measurements. This was done to obtain some protection against Type I errors, as many tests were made.

Furthermore, if a main effect is significant alone and is also included in an important interaction, then discussion is limited to the interaction term alone. While discussion is limited to those terms in the models that were significant for all of the multivariate and univariate analyses, the means and mean-squared errors of the estimates from other terms that were significant in at least the multivariate analysis are included in Appendix C.

Relatively few interactions and main effects were significant at the $\alpha = 0.01$ level in the models for each of levels of the kb_d factor. For the kb_d=-50 level, only the variables in the a1 by a2 by kb_b interaction were significant for each of the models considered. For the kb_d=10 level, two two-way interactions were sufficient to describe all of the terms that were significant in the multivariate and univariate models: the a1 by kb_b and the dsamp by age interactions. For the kb_d=100 level, the significant factors were the kb_b main effect and the a1 by age interaction. For the kb_d=1000 level, only the kb_b main effect was significant in all models. Figures 7.1 through 7.5 contain main effect and interaction plots for these terms. The results are discussed below. As mentioned earlier, other model terms were significant, but not consistently so. These terms, while not discussed in the text, can be examined through the tables printed in Appendix C.

Before examining the results from the models, consider Table 7.8. This table

kb_d	estimate	mean	MSE
-50	no smooth	4.01	0.20
-50	mutation smooth	4.04	0.20
-50	restrictive smooth	4.02	0.15
10	no smooth	2.30	1.15
10	mutation smooth	2.25	0.90
10	restrictive smooth	2.10	0.91
100	no smooth	3.85	1.46
100	mutation smooth	3.98	1.21
100	restrictive smooth	3.77	1.00
1000	no smooth	6.16	1.42
1000	mutation smooth	6.21	1.33
1000	restrictive smooth	6.31	0.79

Table 7.8 Means and mean squared errors from the models fit to the log of the prediction error plus one for different levels of the kb_d factor and the different estimation techniques.

demonstrates that all three estimation procedures behave similarly for each of the levels of kb_d. However, the estimates that were smoothed tended to have smaller values of MSE. They also appear to provide some small benefit when only 10 kilobases separate the markers in the map. One point that stands out from this table is that the estimation error was quite large on average when the disease mutation was located outside of the map of markers, even though it was only 50 kilobases from the first marker.

Figure 7.1 contains plots of the interaction that contains the significant information from the model when the disease locus was -50 kb from the first marker in the map. This interaction involves the distance between the markers and the number of alleles at each of the two marker loci in the map. When there were

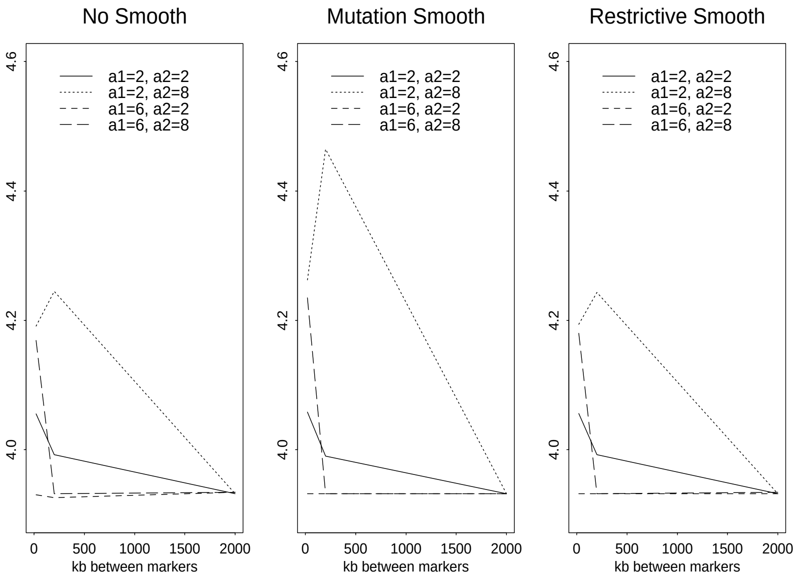

Figure 7.1 Interaction plots for the a1 by a2 by kb_b interaction. The dependent variable is the log of the absolute error plus one, for the case when the disease mutation is -50 kb from the first marker. See Table 7.7 for a description of the factors and their levels.

two alleles at each of the marker loci, the estimation error decreased gradually as the distance between markers was increased. When there were 6 and 8 alleles at marker 1 and 2, respectively, the behavior was similar. However, the error dropped rapidly as kb_b moved from 20 to 200 kb and then leveled off. When there were 2 and 8 alleles at marker 1 and 2, the prediction error changed little. The allele combination that differed the most from the others was when marker 1 had 2 alleles and marker 2 had 8 alleles. In this case, the prediction error increased as kb_b went from 20 to 200 kb before falling when the distance between the two markers reached 2000 kilobases.

All of the allele combinations behaved similarly when a great distance separated the markers. Different allele patterns produces estimates of different quality for reasons that are not clear at this time. It is worth mentioning here, however, that none of the estimators performed particularly well in this case, since the mutation was outside the map of markers. Rather than concern oneself with this result, it would be preferable to type markers that the disease gene is almost sure to be within the map of marker loci.

Figure 7.2 contains plots of one of the two interactions that were important for the case when the disease mutation was 10 kb from the first marker: the a1 by

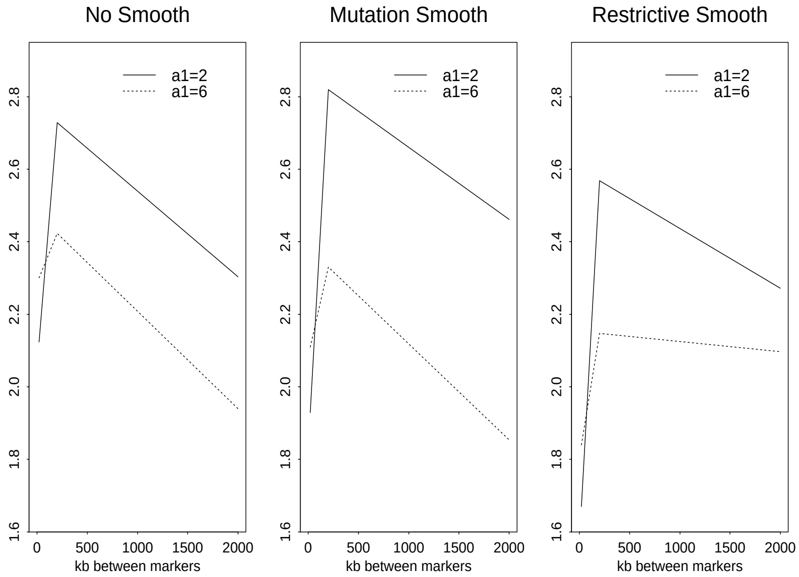

Figure 7.2 Interaction plots for the a1 by kb_b interaction. The dependent variable is the log of the absolute error plus one, when the disease mutation is 10 kb from the first marker. See Table 7.7 for a description of the factors and their levels.

kb_b interaction. This interaction has a simple interpretation. If the marker loci are separated by 10 kb, it is better to use biallelic markers. Otherwise, it is better to use markers that have more than two alleles.

Figure 7.3 contains plots of the other interaction that was significant when the disease mutation was 10 kb from the first marker. One result is clear from this plot: when the sample of disease chromosomes increased, the prediction error tended to decrease. The age of the disease also tended to make the prediction error decrease, but the patterns in which the prediction error decreased depended on sample size. If the sample size was 25 disease chromosomes, moving from 100 to 200 generations

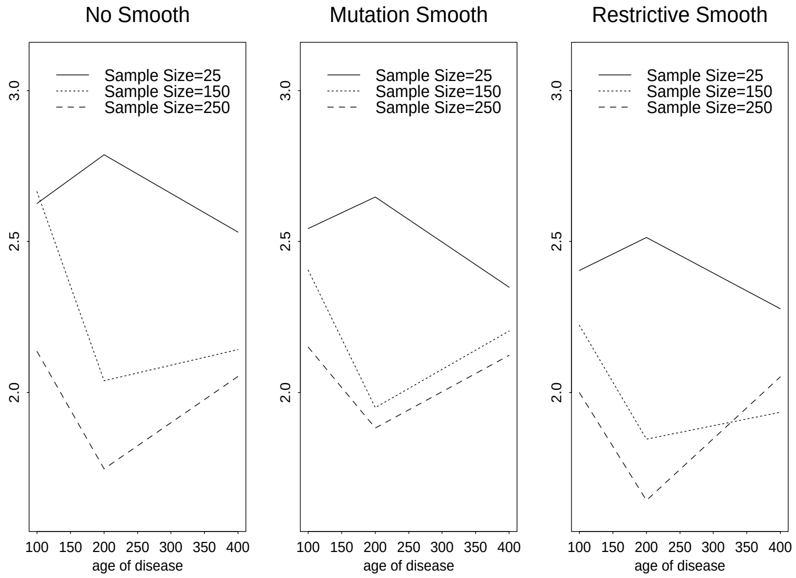

Figure 7.3 Interaction plots for the dsamp by age interaction. The dependent variable is the log of the absolute error plus one, when the disease mutation is 10 kb from the first marker. See Table 7.7 for a description of the factors and their levels.

increased the prediction error to its highest point. On the other hand, if the sample size was larger, an age of 200 generations produced the smallest prediction error.

Figure 7.4 contains plots for the single interaction to be discussed for the case when the disease mutation was 100 kb from the first marker. The case where no smoothing was applied was quite similar to the case when the restrictive smooth was used. In this case, biallelic markers were better then multi-allelic markers when the disease mutation was 100 generations old, but worse when the disease mutation was 200 generations old. The differences were insignificant when the disease mutation was 400 generations old. The most striking feature of this inter-

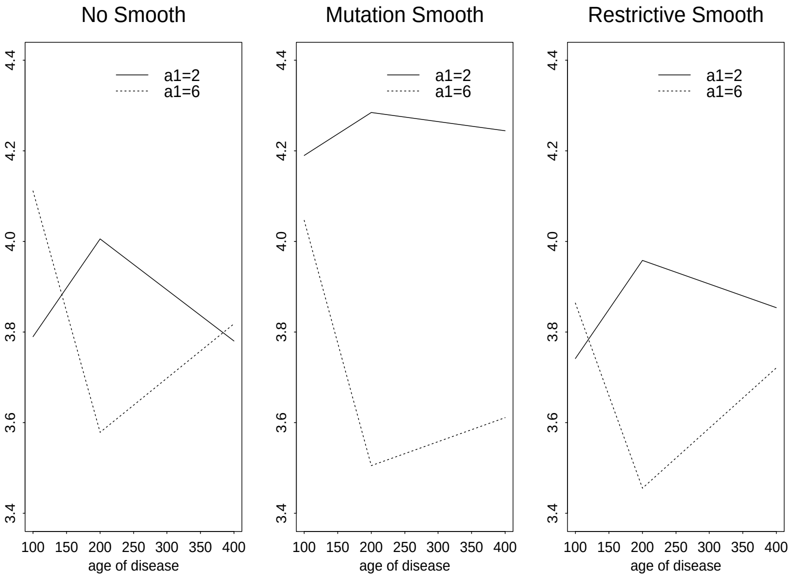

Figure 7.4 Interaction plots for the a1 by age interaction. The dependent variable is the log of the absolute error plus one, when the disease mutation is 100 kb from the first marker. See Table 7.7 for a description of the factors and their levels.

action was that the mutation smooth greatly influenced by the number of marker alleles. When there were only two alleles, it performed poorly. When there were six alleles, it performed well for diseases over 100 generations old.

The plots in Figure 7.5 differ from those in the previous figures. They are main effect plots for the kb_b factor. Also, they display results for two levels of the kb_d factor. They highlight a difference showcased in Table 7.8: the location of the disease mutation relative to the map of marker loci had a big influence on prediction error. Other differences here were slight. The results from the "No Smooth" and "Mutation Smooth" were not significantly different. The "Restrictive Smooth"

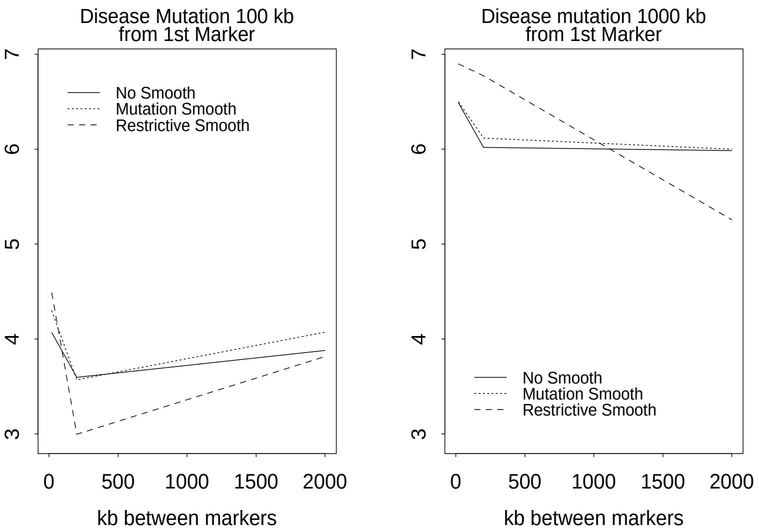

Figure 7.5 Effect plots for the kb_b main effect. The dependent variable is the log of the absolute error plus one, when the disease mutation is 100 kb and 1000 kb from the first marker. See Table 7.7 for a description of the factors and their levels.

estimates were better for the factor level combinations of kb_d=100, kb_b=200 and kb_d=1000, kb_b=2000. However, they were worse when kb_d was 1000 and kb_b was 20.

This brings to light a property of the restrictive smoothing technique. It tends to predict that the disease mutation will lie somewhere near the center of the map of markers. For this reason alone, it would seem to be unwise to use this ad hoc smoothing method.

The second behavior considered through this factorial experiment was the the coverage probability of the approximate intervals. To evaluate the results, a categorical model that used main effects and two-way interactions to predict the coverage probability was fit via maximum likelihood for each estimation method. The maximum likelihood analysis of variance tables are included in Appendix C. The tables indicate that the models fit the data quite well, and perhaps too well, as the p-values for the goodness-of-fit were all approximately equal to one. In keeping with the practice established for interpreting the prediction error, the text discusses only the terms in the model that were significant at the $\alpha = 0.01$ level for all of the smoothing methods. The estimated coverage probabilities for various interactions are included in Appendix C. The plots in Figures 7.6 through 7.10 contain the main effect and interaction plots that are discussed in the text.

Figure 7.6 contains the main effect plot for the factor that influenced coverage probability the most: the location of the disease mutation, or kb_d. The most

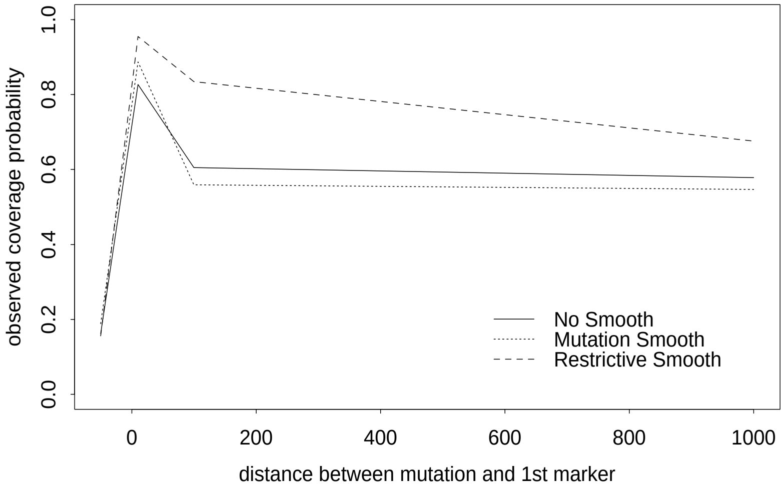

Figure 7.6 Effect plots for the kb_d main effect. The dependent variable is the coverage probability. See Table 7.7 for a description of the factors and their levels.

striking feature of this plot is that the coverage probability was extremely low when kb_d was equal to -50, being less than 20 percent. It climbed above 80 percent when the disease locus was 10 kilobases from the first marker in the map. It then leveled out for larger values of kb_d, at about 60 percent for the "No Smooth" and "Mutation Smooth" methods and at about 70 percent for the "Restrictive Smooth" method.

As had been hoped, the smoothing procedures provided increased coverage probabilities when the disease mutation was separated from the first marker locus by only 10 kilobases.

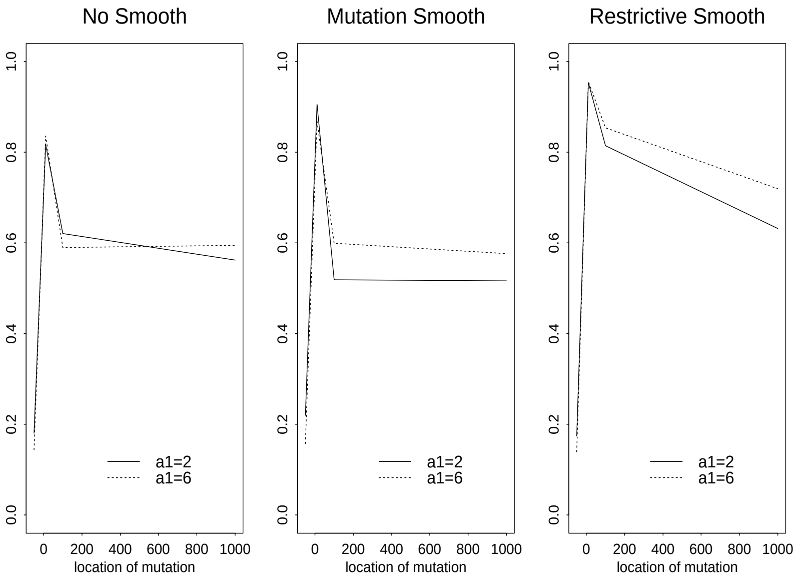

Figure 7.7 Interaction plots for the a1 by kb_d interaction. The dependent variable is the coverage probability. See Table 7.7 for a description of the factors and their levels.

Figure 7.7 contains interaction plots for the kb_d by a1 interaction. The dominant feature is the effect of the levels of the kb_d factor. The a1 factor added some information, however. The two levels of the a1 factor produced similar results when kb_d is small, but as kb_d increased, the coverage probability decreased less for marker loci with 6 alleles than for biallelic markers. This difference was less marked when no smoothing procedure was performed.

The dsamp by kb_d interaction was second only to kb_d in importance when no smoothing was applied. Figure 7.8 contains the interaction plots for this interaction. The first, and perhaps most startling, feature that is apparent from these plots is the fact that lower sample sizes produce better coverage probabilities. This

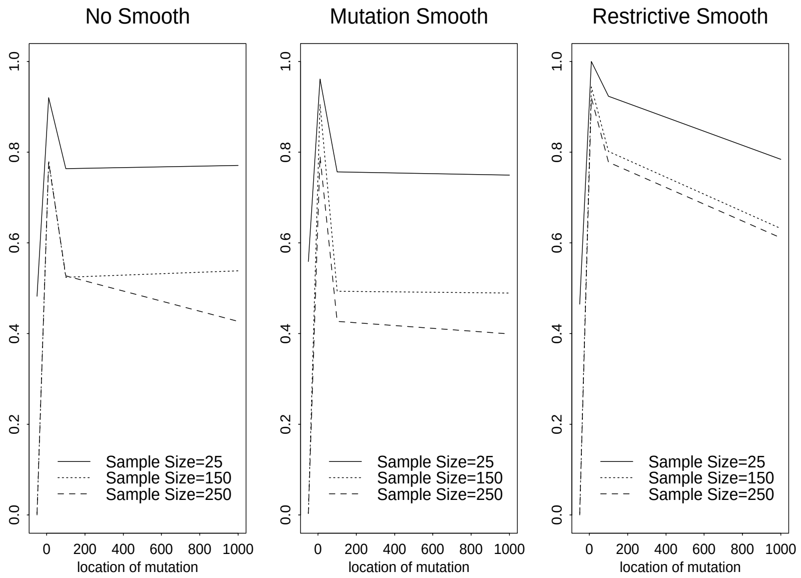

Figure 7.8 Interaction plots for the dsamp by kb_d interaction. The dependent variable is the coverage probability. See Table 7.7 for a description of the factors and their levels.

suggests that the narrower confidence intervals produced by larger sample sizes can be misleading. This also verifies that composite likelihoods do not produce consistent estimators, something mentioned by Devlin *et al.* [12].

Figure 7.9 illustrates the kb_b by kb_d interaction for each of the estimation procedures. The coverage probabilities were highest when the disease mutation was 10 kb from the first marker. When the markers were separated by 20 kb, the coverage probabilities were 50 percent or lower when the disease mutation was more than 20 kilobases from the first marker. This reinforces an observation made in conjunction with the kb_d factor alone. Disease mutations lying outside of the map of marker loci are very difficult to detect.

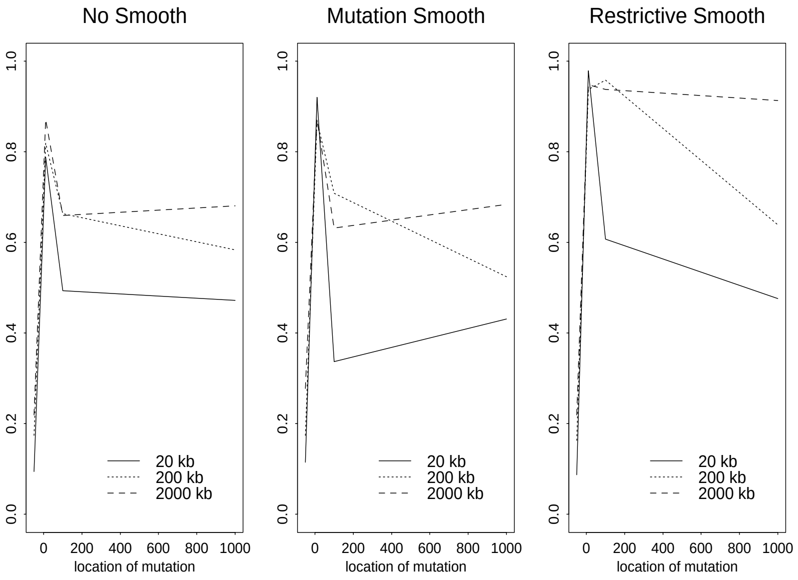

Figure 7.9 Interaction plots for the kb_b by kb_d interaction. The dependent variable is the coverage probability. See Table 7.7 for a description of the factors and their levels.

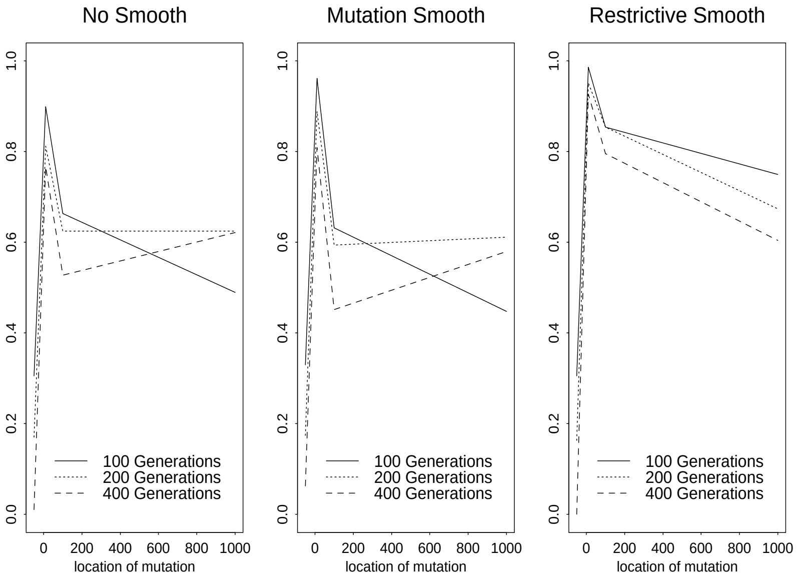

Figure 7.10 Interaction plots for the age by kb_d interaction. The dependent variable is the coverage probability. See Table 7.7 for a description of the factors and their levels.

Two results stand out from the plots in Figure 7.10. The coverage probabilities are quite low for young disease mutations that are far from the markers in the map. Also, the coverage probabilities for disease mutation that are 200 generations old are low when the disease mutation is 100 kb from the first marker in the map. These results are not true for the Restrictive Smooth estimates, which exhibit lower coverage probabilities for older mutations, regardless of the location of the disease mutation, with those for younger mutations decreasing more slowly as the disease location goes from 100 to 1000 kb from the first marker in the map.

In summary, several observations can be made about the results of this simulation. The first is that the location of the disease mutation is the most important factor in controlling the behavior of the composite likelihood estimates. It had a tremendous impact on the prediction errors and on the coverage probabilities as well. This is an unfortunate result. However, closer examination of the results brings hope. The cases where the estimators performed poorly were when the disease mutation was outside the map of marker loci. Using a map of marker loci that is very likely to contain the disease mutation will improve performance. This map may be defined through prior linkage analysis studies or through other methods.

Another result that was common to both the prediction error and coverage probability analyses was that new mutations did not significantly influence the performance of the composite likelihoods. This is likely due to the fact that the variability inherent in the stochastic models for disease propagation dwarfs the variability induced by varying the mutation rates.

Another interesting result is the effect of the dsamp factor. Increasing the sample size decreased the prediction error. However, it decreased the coverage probability as well. The improved prediction error is probably due to the fact that larger sample sizes produce more precise estimates of the marker allele frequencies. This decrease in allele frequency estimation error is reflected in the approximate log likelihoods, making the search regions too narrow.

The final result worth noting here is that the distance between the marker loci had an effect on the estimation. If the marker loci were close together, the prediction errors were small and the coverage probabilities were high as long as the disease mutation was relatively close to one of the marker loci. There is also evidence to suggest that biallelic markers outperform multi-allelic markers in a dense map where the disease mutation is close to a marker locus in a dense map. On the other hand, if the loci are separated by 100 kilobases or more, or if the disease mutation is far from the markers in the map, multi-allelic markers were better.

Age Estimates

When data from several marker loci are available, one can consider estimating the age of the disease in addition to the position of the disease mutation. Figure 7.11

contains a contour plot that jointly estimates the location and age of a disease mutation from a simulated data set. The outer contour is not the lowest point in the composite log likelihood. For visual purposes, the plot looks only at the highest twelve log-likelihood units. The inner-most contour represents an approximate 95% confidence region. Note that locations of the disease gene to the left of the

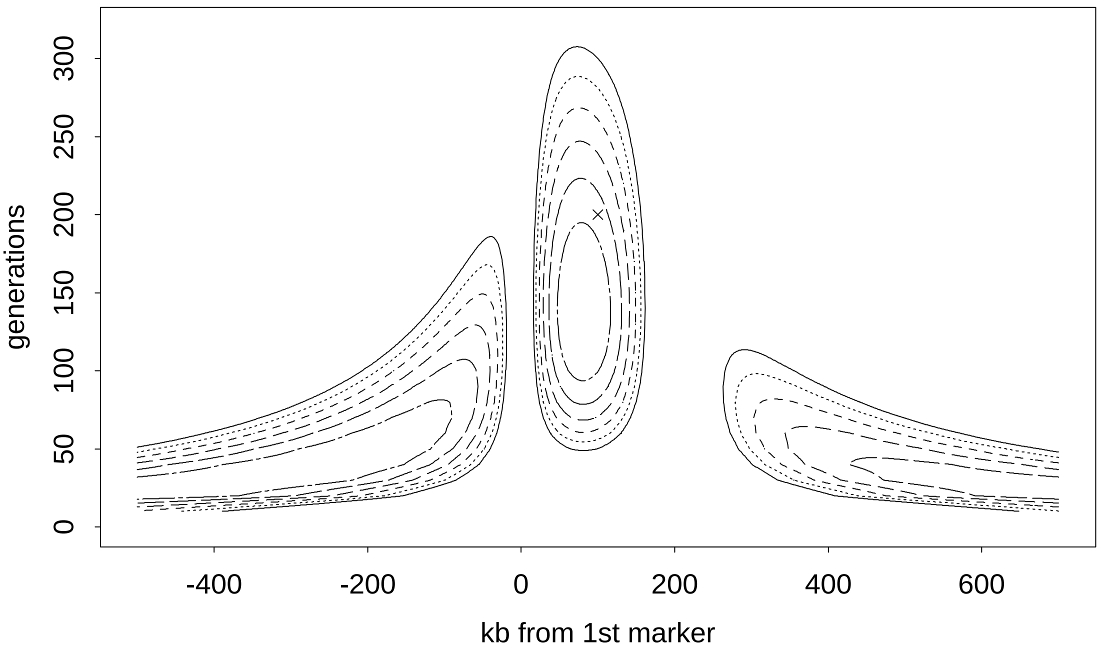

Figure 7.11 Contour plot of the joint composite log likelihood from two-marker simulated data with a sample size of 250. The contours represent two unit increments of the composite log likelihood, with the solid line with the largest circumference representing the lowest value. The truth is marked by ×.

first marker locus cannot be discounted, unless an assumption is made about the minimal possible age of the disease.

The true parameter value lies outside the confidence region defined by the highest contour. It does not miss by much, however. In fact, if age is treated as a nuisance parameter, a confidence region that contains the true location of the disease is defined by projecting the extreme points of the contour to the 'X' axis. This is not the case in a projection to the 'Y' axis, where the truth is still outside the 95% interval. This suggests that estimating the age of a disease is more difficult than estimating its location.

More than Two Marker Loci

This section looks at three situations where data are simulated from a map of ten marker loci. In the first two examples, the disease gene resides in a region densely populated by markers. In the first, the disease is near the end of the map and in the second, the disease is near the center. In the third example, the disease gene lies close to the end of a less dense map spanning 2000 kilobases.

Figures 7.12 7.13 and 7.14 contain contour plots of very restricted views of the resulting composite log likelihoods. The composite likelihoods for the dense maps come remarkably close to predicting the true location of the disease gene. In each case, the true location lies safely within a ~95% confidence interval that spans about 20 kb. However, they fail to provide good estimates of the age of the disease

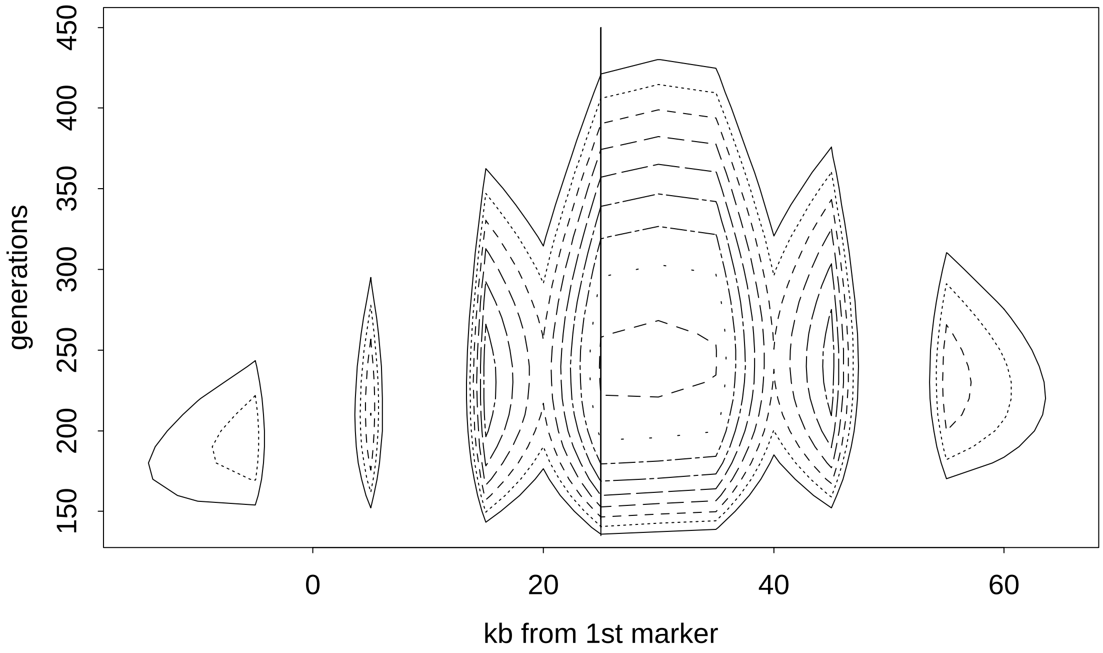

Figure 7.12 Contour plot of a restricted view of the joint composite log likelihood from simulated data with the disease gene near the end of a dense map. The contours represent two unit increments of the composite log likelihood, with the solid line with the largest circumference representing the lowest value. The true location is marked with a vertical line.

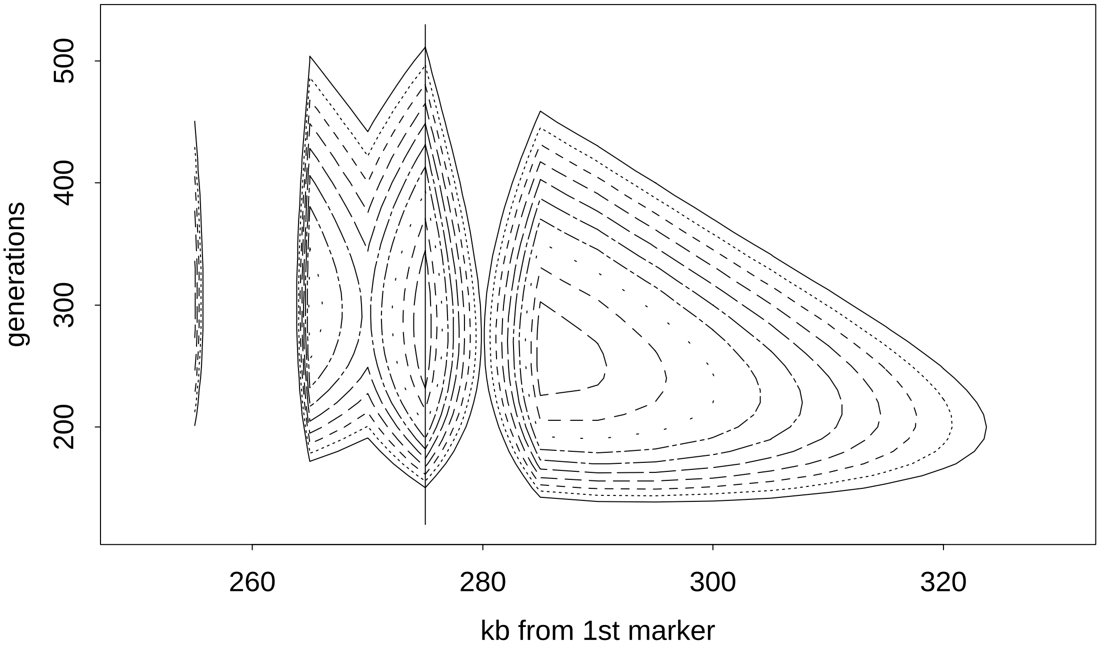

Figure 7.13 Contour plot of a restricted view of the joint composite log likelihood from simulated data with the disease gene near the center of a dense map. The contours represent two unit increments of the composite log likelihood, with the solid line with the largest circumference representing the lowest value. The true location is marked with a vertical line.

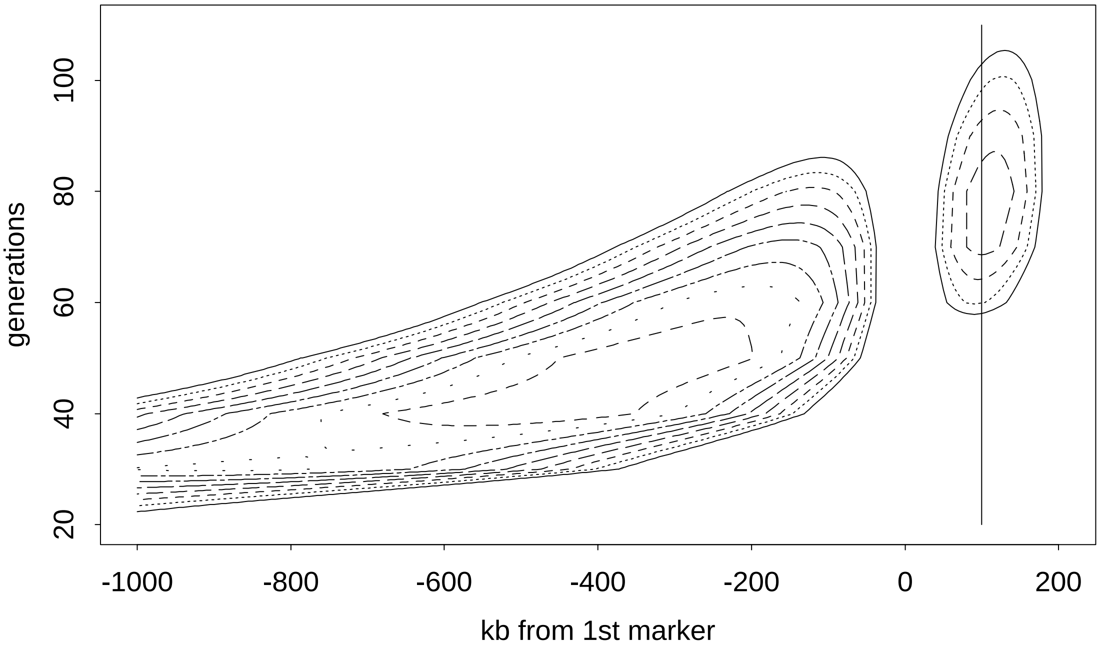

Figure 7.14 Contour plot of a restricted view of the joint composite log likelihood from simulated data with the disease gene near the end of a sparse map. The contours represent two unit increments of the composite log likelihood, with the solid line with the largest circumference representing the lowest value. The true location is marked with a vertical line.

gene, which in each case was 100 generations. This reinforces the result that the two marker haplotypes suggested: it is difficult to estimate the age of a disease mutation.

The result from the sparse map tells a different story. First of all, it comes a little closer in its estimate of the age of the disease. However, the confidence region for the location spans several hundred kilobases and misses the true location by several hundred more. This illustrates two issues. First, in order to map disease genes with high precision, one must have a dense map of genetic markers. Second, if the disease gene lies close to the end of the map of markers, the composite likelihood may have high values outside of the map. This is due to the fact that the component likelihoods have very negative values near marker loci.

Estimates with Mutations at the Marker Loci

When single-marker estimates were made by assuming the mutation matrix in Equation 7.1, the estimates of the distances between markers and disease were reduced. Here, the single-marker likelihoods used to obtain the results listed in Tables 7.3, 7.4 are combined into composite likelihoods. Figure 7.15 contains the composite log likelihood from the first order approximation and Figure 7.16 contains the second order approximation. Note that the composite estimates are not only easier to interpret than the combined single-marker results, but they also appear to provide more reliable estimates of disease location.

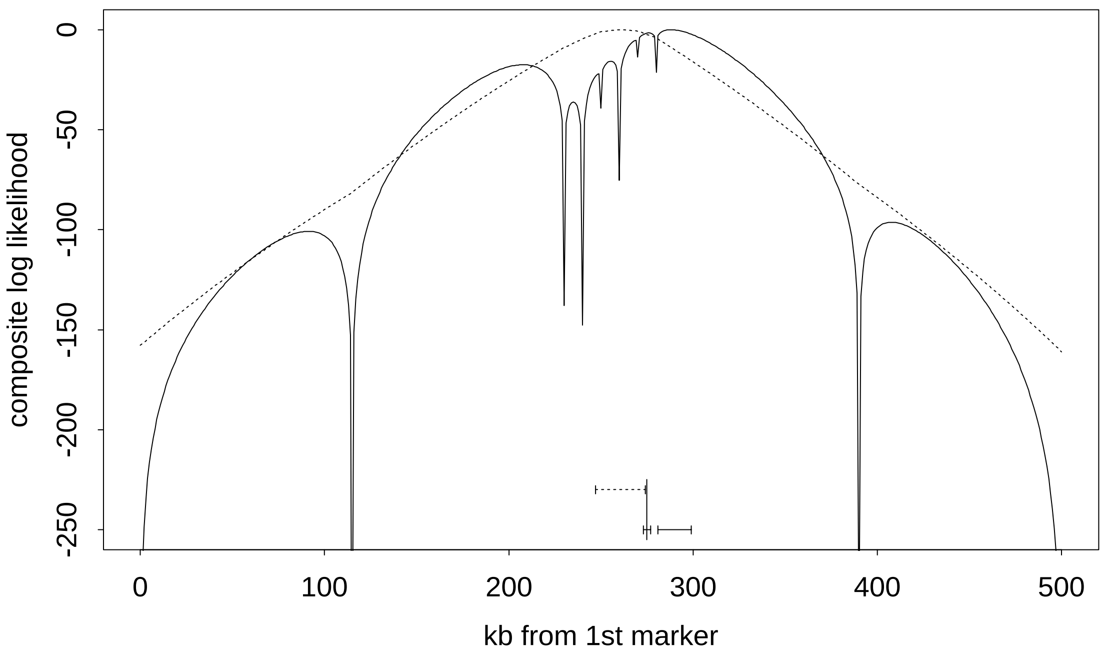

Figure 7.15 Composite log likelihoods with 95% confidence regions comparing mutation and no-mutation estimates for the first order branching process approximation. The solid lines represent the no-mutation estimate and the dashed lines represent the mutation estimate. The true location is marked with a short vertical line.

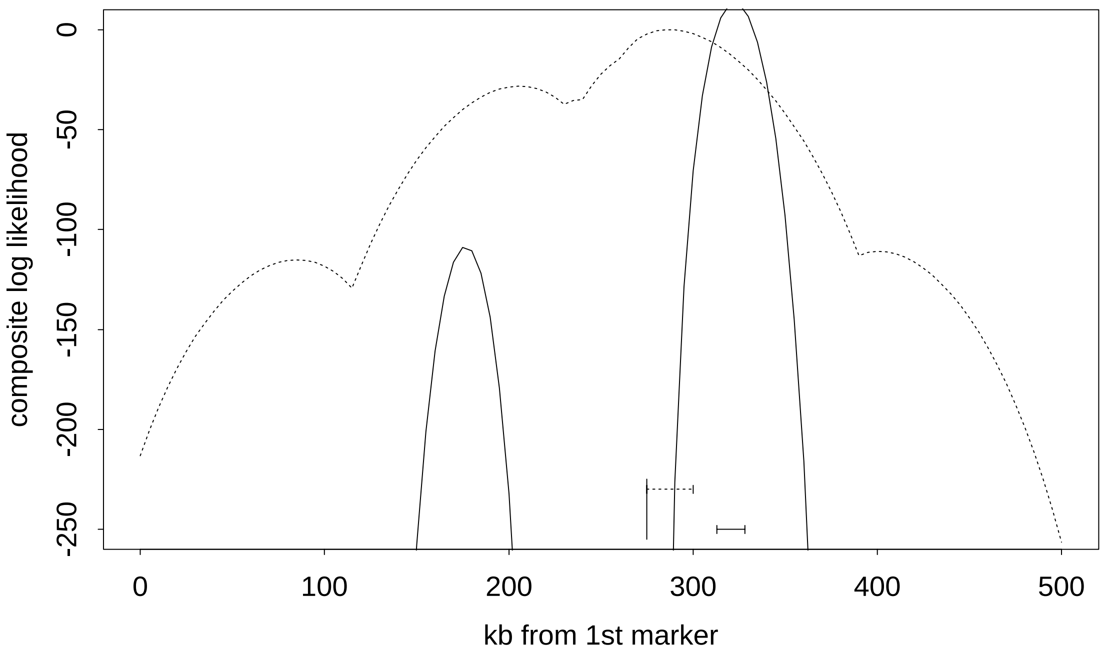

Figure 7.16 Composite log likelihoods with 95% confidence regions comparing mutation and no-mutation estimates for the second order branching process approximation. The solid lines represent the no-mutation estimate and the dashed lines represent the mutation estimate. The true location is marked with a short vertical line.

The composite and single-marker estimates behave quite differently when mutation is used in the estimation procedure when none was present in the evolutionary model. The single marker estimates systematically underestimated the location of the disease gene relative to each marker(see Table 7.5). However, combining those single marker likelihoods into composite likelihoods produced estimates that were quite good. It seems that the composite likelihood is less sensitive to incorrectly specifying mutation processes. This reinforces the result from Chapter 6 which suggested that allowing for mutation acts to smooth the likelihood surface.

For this example, the results from this smoothing-through-mutation are favorable. For each order of approximation, the smoothed composite likelihood suggested a search region of about 30 kb. The first order smoothed interval missed the truth, but only by 1 kb.

The second order smoothed likelihood showed a great improvement over the unsmoothed version. In fact, it defined a search region that captured the true location of the disease gene. This result can be explained by the fact that mutation and recombination parameters are confounded. If mutations occur, then fewer recombinations are required to explain the observed deviation from the initial allele frequencies. This makes it more likely that the disease and marker loci are close together. This increases the value of the log likelihood for small recombination coefficients, helping overcome the instability of the second order approximation to the branching process model.

7.2 Published Data

This section applies the methodology of this thesis to published data and compares the results to those obtained with other methods.

7.2.1 Cystic Fibrosis

Cystic fibrosis (CF) is an autosomal recessive disease that occurs in approximately one of every 2000 live births. It has been estimated that about 70% of all CF chromosomes carry the ΔF_{508} deletion. Due to this strong founder effect, simple linkage disequilibrium mapping techniques aided in the localization of the mutation, which was cloned in 1989 [39].

Kerem *et al.* [39] published data which aided in localizing a disease mutation for Cystic Fibrosis to a region approximately 865 kb from the metD/BanI marker. Using the data published by Kerem *et al.* [39], and assuming that the disease mutation occurred 200 generations ago [36], produces a composite likelihood to estimate the location of the CF disease gene. Figure 7.17 contains two versions of the composite likelihood.

The estimates are relatively close to the truth. The estimate from the no-mutation likelihood misses the truth by 72 kb and the mutation estimate misses the truth by 66 kb. The search regions are more questionable. The simple confidence region defined by the values of the composite log likelihood which are two units less than the maximum are far too narrow. The more conservative intervals are almost

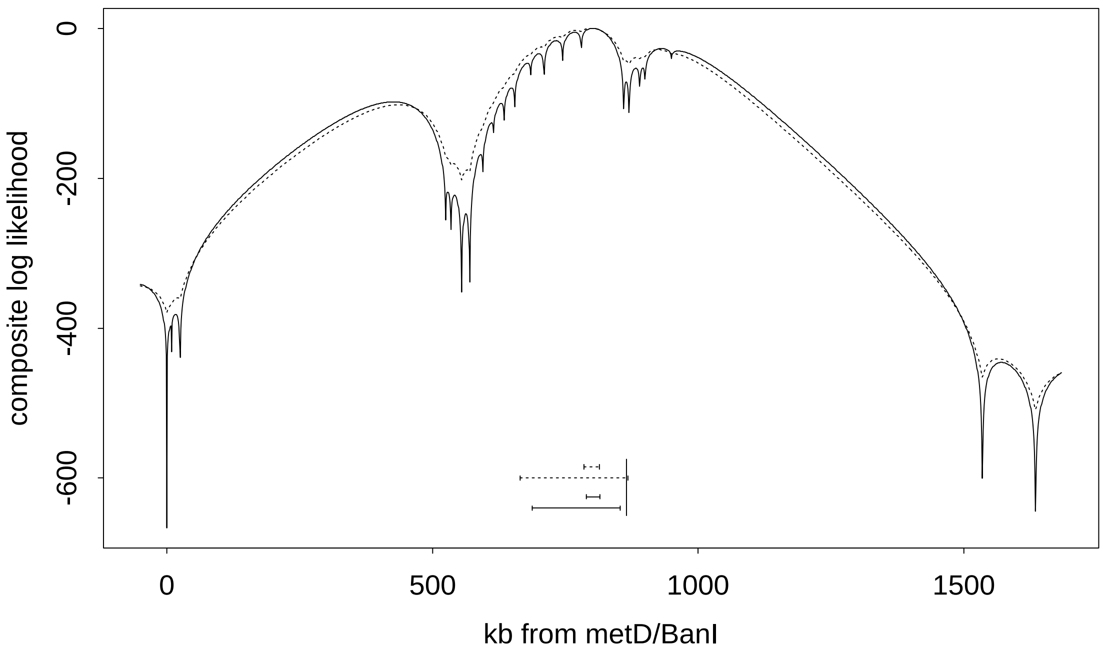

Figure 7.17 Composite log likelihoods comparing mutation ($\mu_d = 0.0001$) and non-mutation estimates of the location of the Cystic Fibrosis gene. The solid lines represent the no-mutation estimate and the dashed lines represent the mutation estimate. Two interval estimates are represented in each case. The shorter was obtained by coming down two log likelihood units from the maximum. The longer interval was constructed by coming down two log likelihood units for each marker included (in this case there were 22). The true location is marked with a short vertical line.

not wide enough. However, these problems can be diminished by jointly estimating the age and location of the disease. For example, in Figure 7.18, the disease gene is located within a region where the joint likelihood maintains relatively high values even for very large ages.

7.2.2 Diastrophic Dysplasia

Diastrophic Dysplasia (DTD) is an autosomal recessive disease that occurs with low frequency in most populations, but with a carrier frequency of 1% - 2% in

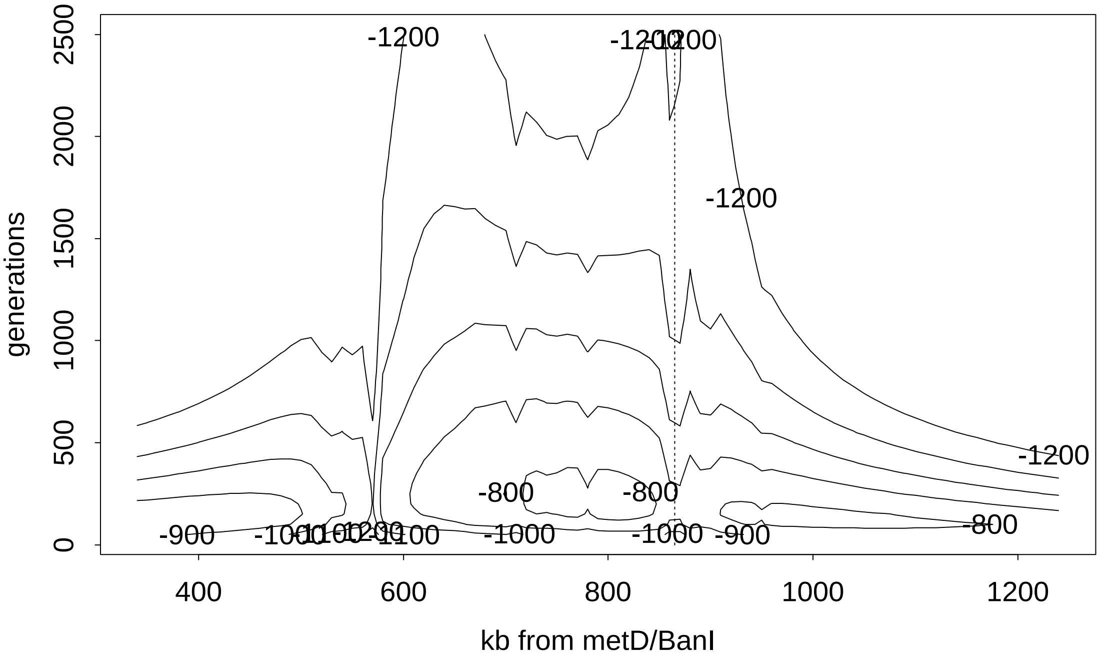

Figure 7.18 Contour plot of the composite log likelihood for the joint estimation of the age and location of the Cystic Fibrosis gene. A vertical dashed line indicates the true location of the ΔF_{508} deletion.

Finland. Hästbacka *et al.* [23] utilized linkage disequilibrium and physical mapping techniques to locate the disease gene.

Using the data published by Hästbacka *et al.* [23], and assuming that the disease mutation occurred 100 generations ago [24], produces composite likelihoods that estimate the known location of the DTD gene. Figure 7.19 contains the composite likelihoods. The unsmoothed likelihood misses the truth by about 95 kb and the conservative confidence interval misses the truth by about 15 kb. The smoothed likelihood reaches its maximum at a point only 50 kb away from the truth, and its conservative confidence interval contains the disease gene.

7.2.3 Huntington's Disease

Huntington's Disease (HD) is an autosomal dominant disease caused by an unstable trinucleotide repeat within a large gene. The disease mutation was identified in 1993 by the Huntington Disease Collaborative Research Group [32].

Using the data published by MacDonald *et al.* [51], makes it possible to form composite likelihoods to predict the location of the disease gene. Since the precise distances were not provided, they were inferred from the locations shown in Figure 2 in the MacDonald *et al.* paper [51]. The inferred distances were in agreement with the few actual distances that were published in the paper. The distances between polymorphisms obtained by digesting the same marker with different enzymes were set equal to 5 kb. The actual distances were uncertain, and ranged

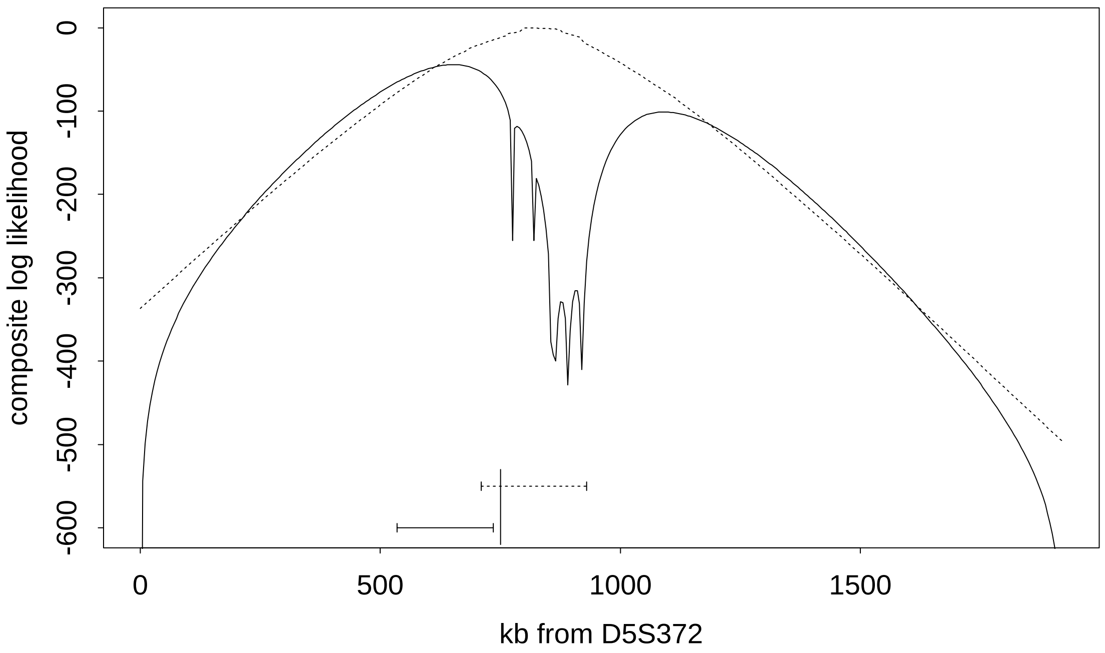

Figure 7.19 Composite log likelihoods comparing unsmoothed and smoothed estimates of the location of the Diastrophic Dysplasia gene. The solid lines represent the no-mutation estimate and the dashed lines represent the mutation estimate. The interval estimates were constructed by coming down two log likelihood units for each marker included, or 20 log likelihood units in this case. The true location is marked with a short vertical line.

from 3 to 20 kb. The likelihoods below assume that the age of the disease is 200 generations, as derived by others (see e.g. [36]).

Figure 7.20 contains the composite log likelihood from the first order approximation to the branching process model. The likelihood upholds the title of the paper from which the data were extracted: there is a complex pattern of linkage disequilibrium in the HD region. This likelihood is does not provide much, if any, refinement to the position of the disease mutation.

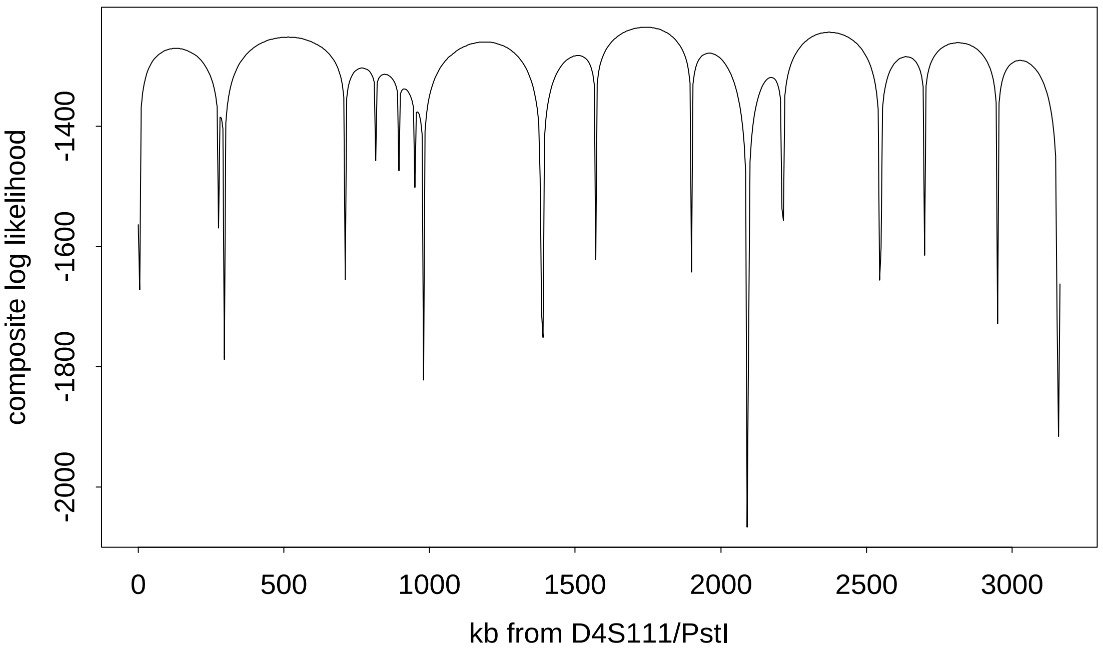

Figure 7.20 Composite log likelihood for the location of the gene for Huntington's Disease.

The reason for the complex pattern of linkage disequilibrium may be due to marker and/or disease mutations. Therefore, a composite likelihood was formed with the same data, but allowing mutational factors to come into play. The mutation effects were introduced in a simple-minded way. Namely, it was assumed that all of the markers share the same rate and pattern of mutation, with a mutation matrix of

$$\begin{pmatrix} 0.999 & 0.001 \\ 0.001 & 0.999 \end{pmatrix} \tag{7.4}$$

It was also assumed that new disease mutations occur at a rate of 1×10^{-8}. The magnitudes of these mutation parameters were roughly based on the estimates of other researchers (see e.g. [76]). The composite log likelihood produced under these assumptions is contained in Figure 7.21. The estimated location of the disease mutation is approximately 2310 kb from the D4S111/*PstI* locus. The resulting estimation error is only 5 kb.

7.2.4 Comparisons

The methods presented in this thesis are not the only composite likelihood techniques. This section compares the estimates from this thesis to to those obtained from other methods.

The method that is the closest to those presented here is that of Xiong and Guo [76]. In fact, when there is no mutation, their first order approximation is

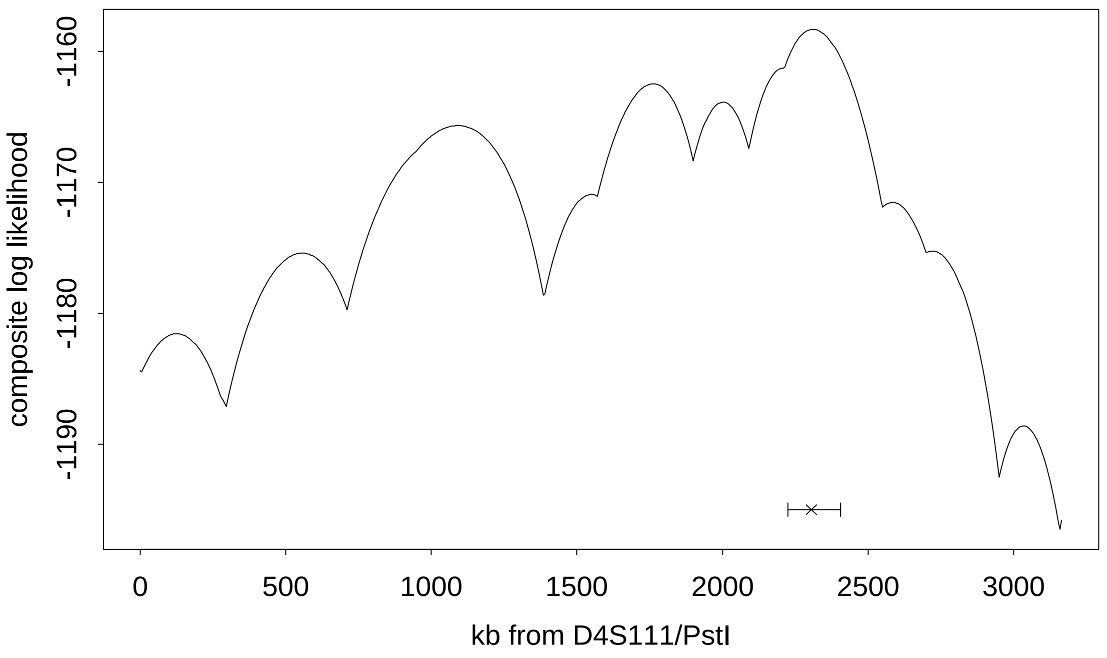

Figure 7.21 Composite log likelihood for the location of the Huntington's Disease gene, assuming that the markers mutate according to the mutation matrix in Equation 7.4, and that the disease gene experiences new mutations at a rate of 1×10^{-8}. The search region is marked at the bottom of the plot, and the true position of the disease mutation is marked by ×.

identical to the first order approximation via the Moran/Coalescent model (see Chapters 3 and 6). Previous results show that the branching process and Moran first order approximations are very similar. Therefore, all estimates obtained using the branching process first order approximation should be very close to the estimates of Xiong and Guo. This is indeed the case. Consider, for example, Cystic Fibrosis. The method of Xiong and Guo predicts the disease gene to lie about 75 kb from the truth, while the branching process first-order approximation has an error of about 72 kb. For each method, the confidence intervals fail to cover the true location of the disease mutation. However, the techniques of smoothing presented in this thesis provide improved estimates of the location of the disease mutation.

Another composite likelihood method is that of Devlin *et al.* [12]. To compare the methods of this thesis to theirs, consider the DTD data of Hästbacka *et al.* [23]. The prediction error from both of their models, simple and heterogeneity, was about 100 kb. Their simple model produced a confidence interval that did not cover the true location, although the heterogeneity model did. This compares well with the results in the previous section, where the recombination-only model failed to produce a correct search region, while the smoothed composite likelihood identified a search region which covered the true location of the disease mutation.

One feature that other composite likelihoods share with those obtained through the methods of this thesis is a propensity to have small values in a region where the map of marker loci is dense. This feature is less marked for the Devlin *et al.* [12] heterogeneity model, although it is still present. Using the methods of this thesis, it is a simple matter to allow for mutation, which can reduce this behavior and even make it disappear.

In their basic forms, all of these methods are based on the assumption that a single disease mutation accounts for a large proportion of the disease chromosomes. However, for each model, methods have been developed to account for deviations from this assumption. These modifications provide insight about which methods may be preferable.

For the method of Devlin *et al.*, there is a heterogeneity parameter to model deviations from the expected behavior. However, the way that it does so is unclear. That is not the case for the methods of this thesis, and that of Xiong and Guo. By using population models to obtain estimators, one can utilize mutation parameters to account for heterogeneity in a way that can be quantified and understood.

From a results standpoint, the estimators from this thesis are quite similar to those of Xiong and Guo. Obtaining the various estimators is vastly different, however. The method of Xiong and Guo requires that a different system of differential equations be solved for each type of mutation matrix. If the mutation pattern is very complicated, it becomes difficult to write the differential equations. With the methods of this thesis, the transition matrix due to both mutation and recombination is a product of mutation and recombination matrices. With this transition matrix, it is a simple matter to obtain the first order approximations, either through taking matrix powers or through obtaining the exponential form of a matrix.

Chapter 8

Conclusion

The slow decay of linkage disequilibrium between loci that are tightly linked suggests that it can be used to obtain refined information about the location of disease genes. This indication is supported by the successes of various researchers (see e.g. [39] [23] and [11].

Methods for linkage disequilibrium mapping can be separated into two major classes: population and family-based methods. Family-based techniques include the transmission disequilibrium test [65] and haplotype relative risk methods [44]. This thesis focuses on disequilibrium mapping with population data.

Population-based methods for linkage disequilibrium can be grouped into two categories. The first of these estimates the location of disease genes based on the relative magnitude of linkage disequilibrium measures among a map of genetic markers. Specific techniques that fit into this group include simple disequilibrium mapping (see e.g. [11] and moment estimators of r (see e.g. [24] [73]). The second provides maximum likelihood estimates of the recombination coefficient through the use of sampling and population models. The contributions of this thesis fall into this class. Other maximum likelihood methods are due to Hill and Weir [29], Kaplan *et al.* [36], Terwilliger [68], Devlin *et al.* [12], Xiong and Guo [76], Rannala and Slatkin [59], and Lange and Fan [46].

Up until very recently, little work had been done to extend linkage disequilibrium mapping outside the framework of simple mapping. Recent work by Terwilliger [68] and Devlin *et al.* [12] developed maximum likelihood procedures that could be used to estimate the recombination coefficient. Their methods were based on sampling and population behavior of specific measures of disequilibrium (primarily δ from Table 3.2).

In 1992, Hästbacka *et al.* [24] made a major contribution by formulating an estimate of recombination fraction for the situation where disequilibrium is present in a young, isolated and rapidly-growing population. This had the effect of making it possible to estimate the location of a disease gene, conditional only on the age of the disease, without specific reference to a measure of disequilibrium.

Kaplan *et al.* [36] made a second contribution by realizing that it is not necessary to model the history of the entire population. Rather, it is sufficient to model the population of disease chromosomes within a large, non-disease population. They proposed that this could be done with a multi-type Galton-Watson branching process with Poisson offspring distributions, which is a rare-disease approximation to the Wright-Fisher model.

Using a multinomial sampling model, they defined the likelihood in Equation 3.12. This likelihood (or log likelihood) could then be used to estimate the location of disease genes. They simulated realizations from the branching process to obtain Monte Carlo estimates of the likelihood.

This simulation procedure required vast amounts of computer time, and its results were subject to potentially large simulation error. This problem prompted the initial work of this thesis. Its aim was to study the behavior of their branching process model in an attempt to improve its utility in estimating r. This led to expressions of the first two moments of the distributions of allele counts, conditional on the age of the disease. Later work also obtained moments of allele frequencies from a time-continuous version of the Moran model via the coalescent. These moments led to approximations to the likelihood for r, which in some special cases yielded closed-form estimates of the recombination coefficient.

As the work of this thesis was progressing, Xiong and Guo [76] published a paper, where they modeled the expansion of a population of disease chromosomes with a diffusion approximation to the Wright-Fisher model. They obtained systems of differential equations that could be used to obtain moment estimates that they used to form approximations to the likelihood. They were the first to describe the technique of approximating the likelihood with low-order Taylor series expansions.

Two papers have recently been published which provide slightly different perspectives. In the first, Lange and Fan [46] generalize several of the assumptions required in existing methods. They assume that the population of normal chromosomes is experiencing deterministic exponential growth, and that new disease mutations occur according to a Poisson point process. After a disease mutation occurs, they model its population behavior with a time continuous branching process. Or specifically, each mutant chromosome lives for an exponential length of time, after which it produces offspring according to some probability law. This probability law can depend on selection pressures, as well as other population parameters. There are three benefits to this technique. First, it is a more reasonable model for disease mutations, as it allows for selective pressures to act against them. Second, it allows for the normal population to be modeled concurrently with the disease population. Third, expectations of a variety of random variables can be obtained by evaluating Laplace transforms. These moments can then be used to form approximations to likelihoods for estimation of r.

In the second paper, Rannala and Slatkin [59] assume that a single disease mutation is propagating within a stable normal population. They also assume that the expansion of disease chromosomes can be described by a continuous time birth-death process similar to that used by Lange and Fan [46]. Their approach is very similar to that of Kaplan *et al.* [36], with only the disease population models differing. However, they present two important advances. First, they obtain transition probabilities that account for mutation and recombination simultaneously. More specifically, they find the transition probabilities as general functions of time. Second, in a major departure from existing methods, they use coalescent-based arguments to obtain the sampling distribution of disease chromosomes. Their approach enjoys increased computational efficiency over that of Kaplan *et al.* [36].

The approach of this thesis is something of a juxtaposition of the work of Kaplan *et al.* [36] and Xiong and Guo [76]. It uses population models, one of which is a Poisson branching process, to derive moments to be used in first and second order approximations to the log likelihood of Equation 3.12. While this technique is closely related to these two, it offers several benefits.

The first order estimator outperformed that of Kaplan *et al.* [36] on data simulated with their branching process recursion. This is a reflection of a problem noted by other researchers, who found that the confidence regions of Kaplan *et al.* [36] can be too wide to be of practical use (see e.g. [76]). Also, the approximations presented in this thesis are much more computationally efficient than the simulation algorithm.

The method of Xiong and Guo [76] is mathematically complex in the sense that systems of differential equations must be obtained and then solved. The differential equations are quite complex, even for the single-step mutation model that the authors propose. Also, in order to use a different model of mutation, one must rewrite the systems of differential equations. Accommodating different mutation patterns invariably makes the system of differential equations more difficult to obtain.

With the methods presented in this thesis, changing the mutation model is as simple as modifying the mutation matrix and multiplying it to the matrix of transition probabilities due to recombination. This produces a transition matrix that can be modified for use in either a branching process or Moran model. The moments necessary to form the approximations can then be obtained through matrix operations on the intensity matrix.

The methods of this thesis perform well, but not perfectly, with respect to data. When used on simulated data, the estimators provided estimates of the location of the disease gene that were close to the truth. However, there were instances when the estimators missed the truth by wide margins. Also, the 95% confidence intervals displayed reduced coverage probabilities in many instances. This is consistent with the results from other methods (see e.g. [12]).

One of the techniques that shows promise in helping overcome the weaknesses of genetic mapping with linkage disequilibrium is the concept of smoothing. Smoothing the composite likelihoods, either by allowing mutation at marker loci or through ad hoc methods, improved the coverage probability on the simulated data. This result was not unexpected. The improvements due to allowing mutation were expected as it is unlikely that recombination alone acts to change marker allele frequencies, even for young diseases propagating in isolated populations. The case for ad hoc methods is stated in Chapter 6. Namely, eliminating extremely negative values near marker loci is extremely valuable when the disease gene is close to a marker locus.

Applying this methodology to several published data sets, produces estimates that agree with the true location of cloned disease genes. The techniques presented in this thesis successfully predicted the location of genes for simple genetic diseases such as Diastrophic Dysplasia and Cystic Fibrosis, as well as for the more complex Huntington's Disease. One important note is that while almost all likelihood-based mapping procedures correctly identify the location of the Cystic Fibrosis and the Diastrophic Dysplasia genes, only those which can incorporate mutations at the disease locus and/or at the marker loci are able to do so with Huntington's disease (see e.g. [36], [12] and [76]).

These results further validate the applicability of linkage disequilibrium mapping as a refinement to the search regions obtained through linkage analysis. Using these methods, the search for disease genes can be narrowed to regions that are much smaller than those obtained through linkage analysis. The estimated intervals were as narrow as 30 kilobases for simulated data, and were several hundred kilobases in length for the real data sets. This compares to intervals from linkage analysis that are on the order of megabases.

8.1 Recommendations

The results of this thesis provide insights to linkage disequilibrium mapping. They lead to several suggestions for those who wish to use it to map disease genes. What follows are several recommendations concerning the location and spacing of marker loci and the selection of population parameters.

8.1.1 Marker Selection

Other researchers have suggested that having markers placed closer together that 60 kb will 'be a waste' [76]. This remark was perhaps precipitated by results from several real data sets. In these data sets, the location of the disease gene was very close to one or more markers. In these cases, the composite likelihood was not maximized at the true location, but rather at some location nearby where no marker was interfering with the signal. The results from this thesis indicate that smoothing the composite likelihood surface can reduce this problem. In fact, using markers that are quite densely spaced can be profitable. This is evidenced in Figure 7.13, where the estimate of disease location was quite close to the truth even within a region populated with markers spaced 10 kb apart. The estimates are even better if the composite likelihood is smoothed, either by accounting for mutation or by using some other method. Other results indicate that if a dense map of markers is to be used, it may be preferable to use markers with few alleles.

Another important aspect to be considered in marker selection is illustrated in Figure 7.14. If the disease gene is located near the end of the dense map of genetic markers, the composite likelihood may peak at a point far outside the map. Smoothing the likelihood can help in this case as well, as it makes the values near markers less negative. However, the problems indicated in the figure were verified in the simulation study. When the disease mutation lies outside of the map of markers, composite likelihood methods do not work well. It is advisable to take cautionary measures to ensure that the disease mutation is likely to be within the map. For example, one could type markers outside each end of the coarse search region defined via linkage analysis.

8.1.2 Population Parameters

It is always preferable to utilize the correct parameters describing population growth, age of disease, mutation, etc. However, if this information is unavailable, one can still obtain reasonable estimates of the location of disease genes.

The first order approximation does not depend on the growth rate, so estimates of disease location can be made without assuming any fixed value. However, the age of the disease and mutation rates and patterns do influence estimates. By combining the information of several markers into a composite likelihood, the impact of these parameters is diminished. Also, if the age of the disease is not known, it can be made a free parameter to be estimated jointly with the disease location.

Treating age as a nuisance parameter in this way can improve estimates. Also, underestimating the age of the disease mutation results in more conservative confidence intervals. When it is clear that mutation is needed for the population model to be reasonable, and no mutation rates are known, using the same mutation rate for all markers seems to work well. This is justified somewhat by the estimates for Huntington's disease in Chapter 7 (see Figure 7.21), where the composite likelihood estimated when the mutation rate and pattern was assumed to be the same for all of the marker loci produced favorable results. Recall that this has the effect of smoothing the composite log likelihood, something beneficial in its own right.

8.2 Future Work

Many issues dealing with linkage disequilibrium mapping remain to be studied. For example, this method assumes that the marker allele frequencies in the entire population are fixed constants. This assumption is not reasonable for markers such as microsatellites which have high mutation rates and/or experience directional mutation. Other population effects such as population substructure, incomplete penetrance and non-rarity of disease may be important factors.

A related issue is the appropriateness of a branching process model for autosomal recessive diseases. Some claim that the correspondence is poor, since individuals must have two copies of the disease allele to contract the disease [46]. As such, Huntington's disease was the only appropriate data set for our branching process model. However, the results from Diastrophic Dysplasia and Cystic Fibrosis, two recessive diseases, were also reasonable. It may be enlightening to study the ability of various population models to approximate different modes of inheritance.

This raises a detail that has been suppressed throughout the thesis. All of the work herein makes the implicit assumption that chromosomes carrying a disease mutation can be distinguished from those that do not. This, in truth, can be quite difficult. For example, if the disease displays an autosomal dominant mode of inheritance, one must decide which of the two copies contains the disease mutation. The difficulties are even more pronounced for complex diseases, where several genes can influence the disease. The results from Huntington's disease are promising in this case. It is an autosomal dominant disease, yet linkage disequilibrium correctly identifies the location of the disease gene by assuming that both of the chromosomes in Huntington's Disease patients carry the disease mutation. Even so, further work needs to be done to address the problem of identifying chromosomes, rather than individuals, that carry a specific disease mutation.

Other interesting paths are laid down by Lange and Fan [46] and Rannala and Slatkin [59]. They present new population and sampling models. More work can be done with generalized population models, including time-continuous branching processes and nonequilibrium normal population models. Also, it is of interest to look into using the coalescent to obtain sampling distributions, thus eliminating the need to evaluate a complex expectation to obtain a likelihood for r.

It is also of interest to move from single-marker to haplotype models. Chapter 6 mentioned some possibilities by considering two-marker transition matrices when there is no mutation. It should be possible to obtain transition matrices that allow for mutation at the marker loci. These transition matrices would make it possible to proceed as in the single marker case to obtain two-marker likelihoods, and even other composite likelihoods, for r.

Another possible project is to study the haplotype population algorithm described in Chapter 7. It may be possible to use it in a way that is analogous to the simulation likelihood technique proposed by Kaplan *et al.* [36].

8.3 Conclusion

This thesis examines the use of stochastic population models relative to linkage disequilibrium mapping. In particular, it presents derivations of approximate likelihood equations based on moment estimates from two specific models: multi-type Galton-Watson branching processes and a time-continuous version of the Moran model. This scheme provides the benefit of being able to easily incorporate any desired model of mutation into the estimation of disease gene location. Results indicate that the techniques presented in this thesis can be used to narrow physical search regions to sequences of DNA that are on the order of several tens to several hundreds of base pairs in length.

Bibliography

[1] Ayala, F.J. and Kiger, J.A. (1984). *Modern Genetics.* Benjamin/Cummings, Menlo Park, Calif.

[2] Besag, J.E. (1974). Spatial interaction and the statistical analysis of lattice systems (with discussion). *J. R. Statist. Soc. B* 36:192-236.

[3] Bishop, Y.M.M.; Feinberg, S.E. and Holland, P.W. (1975). *Discrete Multivariate Analysis: Theory and Practice.* The MIT Press, Cambridge, Mass.

[4] Bradley, G.L. (1975). *A Primer of Linear Algebra.* Prentice-Hall, Inc. Englewood Cliffs, N.J.

[5] Brown, A.H.D.; Feldman, M.W. and Nevo, E. (1980). Multilocus structure of natural populations of *Hordeum spontaneum. Genetics* 96:523-536.

[6] Carter, T.C. and Falconer, D.S. (1951). Stocks for detecting linkage in the mouse and the theory of their design. *J. Genet.* 50:307-323.

[7] Chakraborty, R.; Zhong, Y.; de Andrade, M.; Clemens, P.R.; Fenwick, R.G. and Caskey, C.T. (1994). Linkage disequilibria among $(CA)_n$ polymorphisms in the human dystrophin gene and their implications in carrier detection and prenatal diagnosis in Duchenne and Becker muscular dystrophies. *Genomics* 21:567-570.

[8] Chakraborty, R. and Weiss, K.M. (1988). Admixture as a tool for finding linked genes and detecting that difference from allelic association between loci. *Proc. Natl. Acad. Sci. USA* 85:9119-9123.

[9] Chakraborty, Ranajit (1984). Detection of nonrandom association of alleles from the distribution of the number of heterozygous loci in a sample. *Genetics* 108:719-731.

[10] Chakraborty, Ranajit (1981). The distribution of the number of heterozygous loci in a sample. *Genetics* 98:461-466.

[11] Devlin, B. and Risch, Neil (1995). A comparison of linkage disequilibrium measures for fine-scale mapping. *Genomics* 29:311-322.

[12] Devlin, B.; Risch, N. and Roeder, K. (1996). Disequilibrium mapping: composite likelihood for pairwise disequilibrium. *Genomics* 36:1-16.

[13] Ewens, W.J. (1979). *Mathematical Population Genetics.* Springer-Verlag, Berlin.

[14] Falk, C.T. and Rubinstein, P. (1987). Haplotype relative risks: an easy reliable way to construct a proper control sample for risk calculations. *Ann. Hum. Genet.* 51:227-233.

[15] Feller, William (1968). *An Introduction to Probability Theory and its Applications: Volume I*, 3rd Ed. John Wiley and Sons, Inc., New York.

[16] Fisher, R.A. (1930) *The genetical theory of natural selection*, 1st Ed. Clarendon, Oxford.

[17] Fisher, R.A. (1922). On the dominance ratio. *Proc. Roy. Soc. Edin.* 42:321-431.

[18] Haldane, J.B.S. (1919) The combination of linkage values and the calculation of distances between the loci of linked factors. *J. Genet.* 8:299-309.

[19] Hamada, H.; Petrino, M.G. and Takunaga, T. (1982a). Molecular structure and evolutionary origin of human cardiac muscle actin gene. *Proc. Natl. Acad. Sci. USA* 79:5901-5905.

[20] Hamada, H.; Petrino, M.G. and Takunaga, T. (1982b). A novel repeated element with Z-DNA forming potential is widely found in evolutionarily diverse eukaryotic genomids. *Proc. Natl. Acad. Sci. USA* 79:6465-6469.

[21] Harris, T.E. (1963). *The Theory of Branching Processes.* Prentice-Hall, Inc. Englewood Cliffs, N.J.

[22] Hartl, D.L. and Clark, A.G. (1989). *Principles of Population Genetics.* Sinauer Associates, Inc., Sunderland Mass., second edition.

[23] Hästbacka, J.; de la Chapelle, A.; Mahanti, M.M.; Clines, G.; Reeve-Daly, M.P.; Daly, M.; Hamilton, B.; Kusumi, K.; Trivedi, B.; Weaver, A.; Coloma, A.; Lovett, M.; Buckler, A.; Kaitila, I. and Lander, E. (1994). The diastrophic dysplasia gene encodes a novel sulfate transporter: Positional cloning by fine-structure linkage disequilibrium mapping. *Cell* 78:1073-1087.

[24] Hästbacka, J.; de la Chapelle, A.; Kaitila, I.; Sistonen, P.; Weaver, A. and Lander, E. (1992). Linkage disequilibrium mapping in isolated founder populations: diastrophic dysplasia in Finland. *Nature Genetics* 2:204-211.

[25] Hearne, C.M.; Shosh, S. and Todd, J.A. (1992). Microsatellites for linkage analysis of genetic traits. *Trends in Genet.* 8:288-294.

[26] Hedrick, P.W. (1987). Gametic disequilibrium measures: proceed with caution. *Genetics* 117:331-341.

[27] Hill, W.G. (1974) Disequilibrium among several linked genes in finite populations. I. Mean changes in disequilibrium. *Theor. Pop. Biol.* 5:366-392.

[28] Hill, W.G. and Robertson, A. (1968) Linkage disequilibrium in finite populations. *Theor. Appl. Genet.* 38:226-231.

[29] Hill, W.G. and Weir, B.S. (1994). Maximum-likelihood estimation of gene location by linkage disequilibrium. *Am. J. Hum. Genet.* 54:705-714.

[30] Höglund, P.; Sistonen, P.; Norio, R.; Holmberg, C.; Dimberg, A.; Gustavson, K.-H.; de la Chapelle, A. and Kere J. (1995). Fine Mapping of the congenital chloride diarrhea gene by linkage disequilibrium. *Am. J. Hum. Genet.* 57:95-102.

[31] Hudson, R. R. (1987). Estimating the recombination parameter of a finite population model without selection. *Genetical Research* 50:245-250.

[32] Huntington Disease Collaborative Research Group, The (1993). A novel gene containing a trinucleotide repeat that is expanded and unstable on Huntington's disease chromosomes. *Cell* 72:971-983.

[33] Jeffreys, A.J.; Wilson, V. and Thein, S.L. (1985). Hypervariable 'minisatellite' regions in human DNA. *Nature* 314:67-73.

[34] Jeffreys, A.J.; Royle, N.J.; Wilson, V. and Wong, Z. (1988). Spontaneous mutation rates to new length alleles at tandem-repetitive hypervariable loci in human DNA. *Nature* 332:2788-2801.

[35] Jennings, H.S. (1917). The numerical results of diverse systems of breeding, with respect to two pairs of characters, linked or independent, with special relation to the effects of linkage. *Genetics* 2:97-154.

[36] Kaplan, N.L.; Hill, W.G. and Weir, B.S. (1995). Likelihood methods for locating disease genes in nonequilibrium populations. *Am. J. Hum. Genet.* 56:18-32.

[37] Kaplan, N. and Weir, B.S. (1992). Expected behavior of conditional linkage disequilibrium. *Am. J. Hum. Genet.* 51:333-343.

[38] Karlin, S. and Piazza, A. (1981). Statistical methods for assessing linkage disequilibrium at the HLA-A, B, C loci. *Ann. Hum. Genet.* 45:79-94.

[39] Kerem, B.; Rommens, J.M.; Buchanan, J.A.; Markiewicz, D.; Cox, T.K.; Chakravarti, M.B. and Tsui, L. (1989). Identification of the cystic fibrosis gene: Genetic analysis. *Science* 245:1073-1080.

[40] Kimmel, M.; Pankratz, V.S. and Chakraborty, R. (1996). A moment measure of linkage disequilibrium for microsatellite loci (published abstract). *Am. J. Hum. Genet.* 59S:A31.

[41] Kingman, J.F.C. (1980). *Mathematics of Genetic Diversity.* Society for Industrial and Applied Mathematics, Philadelphia.

[42] Kingman, J.F.C. (1982). On the genealogy of large populations. *J. Appl. Prob.* 19A:27-43.

[43] Klitz, W.; Stephens, J.C.; Grote, M. and Carrington, M. (1995). Discordant patterns of linkage disequilibrium of the peptide-transporter loci within the HLA class II region. *Am. J. Hum. Genet.* 57:1436-1444.

[44] Knapp, M.; Seuchter, S.A. and Baur, M.P. (1993). The haplotype-relative-risk (HRR) method for analysis of association in nuclear families. *Am. J. Hum. Genet.* 52:1085-1093.

[45] Kosambi, D.D. (1944) The estimation of map distances from recombination values. *Ann. Eugen.* 12:172-175.

[46] Lange, K. and Fan R. (1997) Branching processe models for mutant genes in nonstationary populations. *Theor. Pop. Biol.* 51:118-133.

[47] Lehesjoki A.; Koskiniemi M.; Norio R.; Tirrito S.; Sistonen P.; Lander E. and de la Chapelle A. (1993). Localization of the $EPM1$ gene for progressive myoclonus epilepsy on chromosome 21: linkage disequilibrium allows high resolution mapping. *Hum. Mol. Genet.* 2:1229-1234.

[48] Lewontin, R.C. (1988). On measures of gametic disequilibrium. *Genetics* 120:849-852.

[49] Lindsay, B.G. (1988). Composite likelihood methods. *Contemporary Mathematics* 80:221-239.

[50] Luria, S.E. and Delbrück, M. (1943). Mutations of bacteria from virus sensitivity to virus resistance. *Genetics* 28:491-511.

[51] MacDonald, M.E.; Lin, C.; Srinidhi, L.; Bates, G.; Altherr, M.; Whaley, W.L.; Lehrach, H.; Wasmuth, J. and Gusella, J.F. (1991). Complex patterns of linkage disequilibrium in the Huntington Disease region. *Am. J. Hum. Genet.* 49:723-734.

[52] Mode, C.J. (1971). *Multitype Branching Processes: Theory and Applications.* American Elsevier, New York.

[53] Moran, P.A.P. (1958). Random processes in genetics. *Proc. Camb. Phil. Soc.* 54:60-71.

[54] Morgan, T.H. (1928) *The Theory of Genes* Yale University Press, New Haven, Conn.

[55] Morton, N.E. and Wu, D. (1988). Alternative bioassays of kinship between loci. *Am. J. Hum. Genet.* 42: 173-177.

[56] Muller, J. (1916). The mechanism of crossing over. *Am. Nat.* 50:193-207.

[57] Nakamura, Y.; Leppert, M.; O'Connell, P.; Wolff, R.; Holm, T.; Culver, M.; Martin, C.; Fujimoto, E.; Hoff, M.; Kumlin, E. and White, R. (1987). Variable number of tandem repeat (VNTR) markers for human gene mapping. *Science* 235:1616-1622.

[58] Ott, Jurg (1991). *Analysis of Human Genetic Linkage* . The Johns Hopkins University Press, Baltimore, Maryland.

[59] Rannala, B. and Slatkin, M. (1998). Likelihood analysis of disequilibrium mappind, and related problems. *Am. J. Hum. Genet.* 62:459-473.

[60] Rao, D.C.; Morton, N.E.; Lindsten, J.; Hulten, M. and Yee, S. (1977). A mapping function for man. *Human Heredity* 27:99-104.

[61] Renwick, J.H. (1969). Progress in mapping human autosomes. *Br. Med. Bull* 25:65-73.

[62] Resnick, S.L. (1992). *Adventures in Stochastic Processes.* Birkhäuser, Boston, Mass.

[63] Robbins, R.B. (1918). Some applications of mathematics to breeding problems. III. *Genetics* 3:375-389.

[64] Sham, P.C. and Curtis, D. (1995). An extended transmission/disequilibrium test (TDT) for multi-allele marker loci. *Ann. Hum. Genet.* 59:323-336.

[65] Spielman, R.S.; McGinnis, R.E. and Ewens, W.J. (1993). Transmission test for linkage disequilibrium: The insulin gene region and insulin-dependent diabetes mellitus (IDDM). *Am. J. Hum. Genet.* 52:506-516.

[66] Sved, J.A. (1968). The stability of linkedsystems of loci with small population size. *Genetics* 59:543-563.

[67] Tajima, F. (1983). Evolutionary relationship of DNA sequences in finite populations. *Genetics* 105:437-460.

[68] Terwilliger, J.D. (1995). A powerful likelihood method for the analysis of linkage disequilibrium between trait loci and one or more polymorphic marker loci. *Am. J. Hum. Genet.* 56:777-787.

[69] Terwilliger, J.D. and Ott, Jurg (1994). *Handbook of Human Genetic Linkage.* The Johns Hopkins University Press, Baltimore, Maryland.

[70] Thomsom, G. (1981). A review of theoretical aspects of HLA and disease associations. *Theor. Pop. Biol.* 20:168-208.

[71] Tseng, H. and Green, H. (1988). Remodeling of the involucrin gene during primate evolution. *Cell* 54:491-496.

[72] Uhrhammer, N.; Lange, E.; Porras, O.; Naeim, A.; Chen, X.; Sheikhavandi, S.; Chiplunkar, S.; Yang, L; Dandekar, S.; Liang, T.; Patel, N.; Teraoka, S.; Udar, N.; Calvo, N.; Concannon, P.; Lange, K. and Gatti, R. A. (1995). Sublocalization of an ataxia-telangiectasia gene distal to D11S384 by ancestral haplotyping in Costa Rican families. *Am. J. Hum. Genet.* 57:103-111.

[73] Wakeley, John (1997). Using the variance of pairwise differences to estimate the recombination rate. *Genetical Research* 69:45-48.

[74] Woolf, B. (1955). On estimating the relation between blood group and disease. *Ann. Hum. Genet.* 19:251-253.

[75] Wright, S. (1931). Evolution in Mendelian populations. *Genetics* 16:97-159.

[76] Xiong, M. and Guo, S. (1997). Fine-scale genetic mapping based on linkage disequilibrium: theory and applications. *Am. J. Hum. Genet.* 60:1513-1531.

Appendix A

Programs

A.1 Recombination-Only Model: FOA

A.1.1 Galton-Watson Branching Process

```
c        07 July 1997
c
c        This program calculates the point and interval estimates from
c        the likelihood described in Chapter 5.
c
c
        implicit none
        integer maxall                  !the maximum number of alleles

        parameter (maxall=25)           !set the maxall parameter

        integer k                       !the actual number of marker alleles
        real*8 disease(maxall)          !disease allele counts
        real*8 normal(maxall)           !normal counts
        integer order(maxall)           !vector containing indices that assure
                                        ! that the most frequent disease allele
                                        ! is in the proper position.
        real*8 totd                     !disease sample size
        real*8 totn                     !normal sample size
        real*8 age                      !age (in generations) of disease
        real*8 pn(maxall)               !freq. of allele 1 in normal pop.
        real*8 pd(maxall)               !observed freq of disease allele 1
        real*8 ests(3)                  !contains the maximum likelihood
                                        !  estimate of r, plus approx 95% int
        integer i                       !loop counter
        character*1 check               !checks inputs

        check = 'N'

        do while (check .ne. 'Y')
          write(*,'(/,A)') 'Enter the number of marker alleles.'
          read(*,*) k
          write(*,'(/,A)') 'Enter the sampled disease allele counts.'
          read(*,*) (disease(i), i = 1, k)
          write(*,'(/,A)')
     1       'Enter the normal allele counts in the same order.'
          read(*,*) (normal(i), i=1,k)
          write(*,'(/,A)') 'Enter the age of the disease (generations).'
          read(*,*) age

          write(*,'(//,A,/)') 'You have entered:'

          write(*,'(a7,15i5)') ' ', (i,i=1,k)
          write(*,'(a8,15f5.0)') 'Disease', (disease(i),i=1,k)
          write(*,'(a8,15f5.0)') 'Normal', (normal(i),i=1,k)

          write(*,'(/,A,f10.0)') 'Age of disease:', age

          write(*,'(//,A)') 'Is that correct? (Y/N)'
          read(*,'(A)') check
          if (check .eq. 'y') check = 'Y'
        enddo

c  Initialize the orderings.

        do i = 1, k
          order(i) = i
        enddo

        call assume(maxall, k, disease, order)

        totn = 0.0d0
        totd = 0.0d0
        do i = 1, k
          totn = totn + normal(i)
          totd = totd + disease(i)
        enddo
        do i = 1, k
          pd(i) = disease(i)/totd
          pn(i) = normal(i)/totn
        enddo

        ests(1) = 1.0d0 - ( (pd(order(1)) - pn(order(1))) /
     1                        (1.0d0 - pn(order(1))) )**(1/age)

        ests(2) = 0.0000000010d0
        ests(3) = 2.50d0 * ests(1)

        call nrlim(disease(order(1)), totd, pd(order(1)),
     1              pn(order(1)), age, ests(2))
        call nrlim(disease(order(1)), totd, pd(order(1)),
     1              pn(order(1)), age, ests(3))

        write(*,*)
        write(*,'(/,A,f11.8)')
     1     'The maximum likelihood estimate of r is:   ', ests(1)
        write(*,'(/,A,e10.4,A,e10.4,A)')
     1            'Approximate 95% confidence interval:  (',
     2                       ests(2),', ',ests(3),')'

        end



        subroutine assume(maxall, k, disease, order)
c
c        This subroutine sorts the alleles by the frequency in which they
c        occur in the disease sample.  This makes it possible to ensure
c        that the assumption that the disease mutation occurred with
c        the most frequent disease allele is used correctly.
c
        implicit none

        integer maxall, k
        real*8 disease(maxall)                !the disease allele counts
        integer order(maxall)                 !the sorted index
        integer i                             !counting variable
        integer ip1                           !i + 1
        integer istop                         !limit of the loop
        integer temp                          !temporary storage
        logical sorted                        !data sorted flag


c  Sort the order array

        istop = k - 1
        sorted = .false.
        do while (.not. sorted)
          sorted = .true.
          do i = 1, istop
            ip1 = i+1
            if (disease(order(i)) .lt. disease(order(ip1))) then
              temp = order(i)
              order(i) = order(ip1)
              order(ip1) = temp
              sorted = .false.
            endif
          enddo
          istop = istop - 1
        enddo
        return
        end



        subroutine nrlim(n1d, totd, p1d, p1n, age, rlim)
c
```

```
c        This subroutine finds the approximate maximum likelihood
c        confidence limits for the recombination coefficient (95%).
c        It is an  application of the Newton-Rhapson algorithm.
c

        real*8  n1d             !number of most frequent disease marker
                                !  alleles (sample size)
        real*8  totd            !complete disease sample size
        real*8  p1d             !the allele frequency of the most
                                !  common disease allele in disease
                                !  sample
        real*8  p1n             !the allele frequency of the most
                                !  common disease allele in normal
                                !  sample
        real*8  age             !age in generations of the disease
        real*8  rlim            !root of the equation (solution)
        real*8  r0, r1          !the iterated values of r
        real*8  a, b, c, d, e, ei  !some temporary quantities
        real*8  num, den        !numerator and denominator of NR update
        real*8  error           !N-R approximation error at each step

        a = n1d*log(p1d)
        c = totd - n1d
        b = c*log(1.0d0 - (p1d-p1n)/(1.0d0-p1n))
        d = 1.0d0 - p1n
        e = n1d*d

        r0 = rlim
        r1 = 1.0d0
        error = 1.0d0

        do while (error .gt. 0.00000001d0)
          ei = (1.0d0 - r0)**age
          num = n1d*log(p1n + d*ei) - a + c*log(1-ei) - b + 2
          den = age*ei/(1.0d0-r0) * (c/(1.0d0-ei) - e/(p1n+d*ei))
          r1 = r0 - num/den
          error = abs(r1-r0)
          r0 = r1
        enddo
        rlim = r1

        return
        end
```

A.1.2 Moran/Coalescent Process

```
c        Expected Moran/Coalescent Likelihood       Shane, 13 January 1998
c
c        This program calculates the likelihood described in the PhD thesis
c        of Shane Pankratz (Moran/Coalescent, Recombination Only).
c
c        The the disease allele frequencies are obtained from a time-continuous
c        Moran model using a coalescent argument (they are expected disease
c        allele frequencies).
c
c
        implicit none
        integer maxall                  !the maximum number of alleles

        parameter (maxall=25)           !set the maxall parameter

        integer k                       !the actual number of marker alleles
        real*8 disease(maxall)          !disease allele counts
        real*8 normal(maxall)           !normal counts
        integer order(maxall)           !vector containing indices that assure
                                        ! that the most frequent disease allele
                                        ! is in the proper position.
        real*8 totd                     !disease sample size
        real*8 totn                     !normal sample size
        real*8 age                      !age (in generations) of disease
        real*8 pn(maxall)               !freq. of allele 1 in normal pop.
        real*8 pd(maxall)               !observed freq of disease allele 1
        real*8 sortpn(maxall)           !sorted normal allele freqs
        real*8 sortnd(maxall)           !sorted disease allele counts
        real*8 ests(3)                  !contains the maximum likelihood
                                        !  estimate of r, plus approx 95% int
        integer i                       !loop counter
        character*1 check               !checks inputs

        check = 'N'

        do while (check .ne. 'Y')
          write(*,'(/,A)') 'Enter the number of marker alleles.'
          read(*,*) k
          write(*,'(/,A)') 'Enter the sampled disease allele counts.'
          read(*,*) (disease(i), i = 1, k)
          write(*,'(/,A)')'Enter the normal allele counts in the same order.'
          read(*,*) (normal(i), i=1,k)
          write(*,'(/,A)') 'Enter the age of the disease (generations).'
          read(*,*) age

          write(*,'(//,A,/)') 'You have entered:'
          write(*,'(a7,15i5)') ' ', (i,i=1,k)
          write(*,'(a8,15f5.0)') 'Disease', (disease(i),i=1,k)
          write(*,'(a8,15f5.0)') 'Normal', (normal(i),i=1,k)

          write(*,'(/,A,f10.0)') 'Age of disease:', age

          write(*,'(//,A)') 'Is that correct? (Y/N)'
          read(*,'(A)') check
          if (check .eq. 'y') check = 'Y'
        enddo

c  Initialize the orderings.
        do i = 1, k
          order(i) = i
        enddo
        call assume(maxall, k, disease, order)

        totn = 0.0d0
        totd = 0.0d0
        do i = 1, k
          totn = totn + normal(i)
          totd = totd + disease(i)
        enddo
        do i = 1, k
          pd(i) = disease(i)/totd
          pn(i) = normal(i)/totn
        enddo

        do i = 1, k
          sortpn(i) = pn(order(i))
          sortnd(i) = disease(order(i))
        enddo

        ests(1) = -1.0d0 / age * dlog(
     1    (pd(order(1)) - pn(order(1))) / (1.0d0 - pn(order(1))) )

        ests(2) = 0.0000000010d0
        ests(3) = 0.50d0

        call nrlim(maxall,k,sortnd,totd,sortpn,age,ests(1),ests(2))
        call nrlim(maxall,k,sortnd,totd,sortpn,age,ests(1),ests(3))

        write(*,*)
        write(*,'(/,A,f11.8)')
     1     'The maximum likelihood estimate of r is:    ', ests(1)
        write(*,'(/,A,e10.4,A,e10.4,A)')
     1            'Approximate 95% confidence interval:  (',
     2                       ests(2),', ',ests(3),')'

        end
```

```
      subroutine assume(maxall, k, disease, order)
c
c        This subroutine sorts the alleles by the frequency in which they
c        occur in the disease sample.  This makes it possible to ensure that
c        the assumption that the disease mutation occurred with the most
c        frequent disease allele is used correctly.
c
      implicit none

      integer maxall, k
      real*8 disease(maxall)                  !the disease allele counts
      integer order(maxall)                   !the sorted index
      integer i                               !counting variable
      integer ip1                             !i + 1
      integer istop                           !limit of the loop
      integer temp                            !temporary storage
      logical sorted                          !data sorted flag


c  Sort the order array

      istop = k - 1
      sorted = .false.
      do while (.not. sorted)
        sorted = .true.
        do i = 1, istop
          ip1 = i+1
          if (disease(order(i)) .lt. disease(order(ip1))) then
            temp = order(i)
            order(i) = order(ip1)
            order(ip1) = temp
            sorted = .false.
          endif
        enddo
        istop = istop - 1
      enddo
      return
      end
      subroutine nrlim(maxall,k,nd,totd,pn,age,rhat,rlim)
c
c      This subroutine finds the approximate maximum likelihood confidence
c      limits for the recombination coefficient (95%).  This routine applies
c      a bisection algorithm.
c
      implicit none

      integer maxall                  !maximum alleles allowed
      integer k                       !number of alleles

      real*8  nd(maxall)              !counts  of sampled disease markers
      real*8  totd                    !complete disease sample size
      real*8  pn(maxall)              !allele frequencies in the normal pop
      real*8  age                     !age in generations of the disease
      real*8  rhat                    !mle
      real*8  rlim                    !confidence limit
      real*8  limit(3)                !the limits for the bisection
      real*8  fx(3)                   !the functional evaluations of limit
      real*8  a, b, c, ei             !some temporary quantities-to save time
      real*8  max                     !max value of the function
      real*8  error                   !approximation error at each step
      integer i                       !loop counter

      a = (totd - nd(1)) * age
      b = 1.0d0 - pn(1)
      c = age * nd(1) * b

      ei = dexp(-rhat*age)
      max = nd(1)*dlog(pn(1) + b*ei)
      do i = 2, k
        max = max + nd(i) * dlog(pn(i) * (1.0d0 - ei))
      enddo

      limit(1) = rhat
      limit(3) = rlim
      if (limit(1) .gt. limit(3)) then
        limit(2) = limit(3)
        limit(3) = limit(1)
        limit(1) = limit(2)
      endif
      limit(2) = (limit(1) + limit(3))/2.0d0
      do i = 1, 3
        call logl(maxall,k,nd,totd,pn,age,max,limit(i),fx(i))
      enddo
      error = dabs(limit(3) - limit(1))
```

```
do while (error .gt. 0.0000000010d0)
  if ((fx(1) .lt. 0.0d0) .and. (fx(2) .gt. 0.0d0)) then
    limit(3) = limit(2)
    limit(2) = (limit(1) + limit(3))/2.0d0
  elseif ((fx(2).lt.0.0d0) .and. (fx(3).gt.0.0d0)) then
    limit(1) = limit(2)
    limit(2) = (limit(1) + limit(3))/2.0d0
  elseif ((fx(2).gt.0.0d0) .and. (fx(3).lt.0.0d0)) then
    limit(1) = limit(2)
    limit(2) = (limit(1) + limit(3))/2.0d0
  elseif ((fx(1).gt.0.0d0) .and. (fx(2).lt.0.0d0)) then
    limit(3) = limit(2)
    limit(2) = (limit(1) + limit(3))/2.0d0
  endif
  do i = 1, 3
    call logl(maxall,k,nd,totd,pn,age,max,limit(i),fx(i))
  enddo
  error = dabs(limit(3) - limit(1))
enddo
rlim = limit(2)

return
end
```

```
        subroutine logl(maxall, k, nd, totd, pn, age, max, r, fx)
c
c This subroutine calculates the value of the log-likelihood,
c scaled such that the mle has a value of 2 (this allows us to
c find the roots of the likelihood where the value of the
c likelihood is 2 less than the maximum.
c
        implicit none

        integer maxall                !maximum alleles allowed
        integer k                     !number of alleles

        real*8  nd(maxall)            !counts  of sampled disease markers
        real*8  totd                  !complete disease sample size
        real*8  pn(maxall)            !allele frequencies in the normal pop
        real*8  age                   !age in generations of the disease
        real*8  a, b, c, ei           !some temporary quantities-to save time
        real*8  max                   !max value of the function
        real*8  r                     !recombination value for function eval
        real*8  fx                    !loglikelihood(r) - max + 2
        integer i                     !loop counter

        a = (totd - nd(1)) * age
        b = 1.0d0 - pn(1)
        c = age * nd(1) * b


        ei = dexp(-r*age)
        fx = nd(1)*dlog(pn(1) + b*ei) - max + 2.0d0
        do i = 2, k
          fx = fx + nd(i) * dlog(pn(i) * (1.0d0 - ei))
        enddo

        return
        end
```

A.2 General Composite Likelihood Program

```
        program composite
c
c       This program calculates composite likelihoods, based on moment
c       estimates (first- and second-order) obtained using branching
c       process or Moran/coalescent models.
c
c===============================================================================
c
c       Variable Declarations
c
        implicit none

        integer maxmark                 !maximum number of markers
        integer maxall                  !maximum number of alleles at each mark
        real*8  conf                    !constant that gives confidence level
                                        !  2 corresponds to ~95%

        parameter (maxmark=30, maxall=10, conf = 2.0d0)

        character*20 filein             !name of the input file
        character*20 fileout            !name of the output file
        character*40 name               !name of the disease
        real*8 dpopsize                 !current size of disease population
        real*8 dismut                   !disease mutation rate
        real*8 lambda                   !population growth rate
        integer nummark                 !number of markers
        character*16 markname(maxmark)  !names of markers
        integer numall(maxmark)         !number of alleles at each marker
        real*8 nd(maxmark,maxall)       !sampled disease chromosomes
        real*8 nn(maxmark,maxall)       !sampled normal chromosomes
        real*8 kb(maxmark)              !kilobases between adjacent markers
        real*8 pd(maxmark,maxall)       !disease marker allele frequencies
        real*8 pn(maxmark,maxall)       !normal marker allele frequencies
        real*8 mutmark(maxmark,maxall,maxall)  !marker mutation matrices
        real*8 agelim(3)                !years to consider
        real*8 kboff                    !distance in kb to calculate the
                                        !  likelihood from the ends of the map
        real*8 kbmesh                   !grid coarseness for the likelihood
        character*1     model   !Population model to be used
                                !  B = Branching Process, M = Moran/Coalescent
        character*1     approx  !Type of approximation to the likelihood
                                !  1 = FOA,  2 = SOA
        character*1     liketype        !Type of likelihood to form
        character*1     smooth          !smooth  'Y' or 'N'
        real*8          coval           !cutoff falue for smoothing
        character*1     input           !when input = I, input the data
        character*1     run             !done with setup (run program when R)
        real*8          age             !age of disease (loop counter)
        real*8          dloc            !disease location (loop counter)
        real*8          llike           !log-likelihood value
        real*8          r               !recombination coefficient
        real*8          mean(maxall,maxall)     !one-step mean matrix
        real*8  mt(maxall,maxall)         !age-generation mean matrix
        real*8  mt_1(maxall,maxall)       !more powers of mean
        real*8  SVE(maxall,maxall)        !sum of V_i*EX_i^(t-1)

        real*8   hessian(maxall,maxall) !the Hessian of prod(pid^nid)
        real*8   EXXt(maxall,maxall)    !the matrix of second moments for
                                        !   the counts
        real*8   EPd(maxall)            !the expected disease allele freqs
                                        !  (need to be in sorted order)
        real*8  hE(maxall,maxall)       !hessian * EXXt

        integer ranks(maxmark,maxall)   !the ranks of the disease allele counts
                                        !   (for each marker separately)
        real*8  Rt(maxall, maxall)      !the joint probs from Moran/Coal

        real*8  foa                     !function that calculates first-order
        integer i !,j,k                 !loop counters
        real*8  temp, temp2             !temporary variables (various uses)
        real*8  dlhat                   !estimate for location of disease locus
        real*8  rhat                    !estimated recombination coef
        real*8  ul, ll                  !upper and lower confidence limits
        real*8  maxlike                 !maximum value of the log likelihood

        data model, approx, liketype, smooth /'B','1','C','N'/
c
c==============================================================================
c
c       Begin Execution

        filein = 'composite.dat'
        fileout = 'composite.out'

        input = 'q'

        do while (input .ne. 'I')
          call menu(model, approx, liketype, kbmesh, kboff,
     1              agelim, dpopsize, smooth, coval, filein,
     2              fileout, input, run)
          if (run .eq. 'Q') goto 9999       !exit program
          if (run .ne. 'R') then
            call datain(maxmark,maxall,filein,name,dpopsize,dismut,
     1                  nummark, markname, numall, nd, nn, kb, pd,
     2                  pn, mutmark, agelim, kboff, kbmesh, input)
          endif
```

```
   enddo

   call rankeach(maxmark,nummark,maxall,numall,nd,ranks)

   if (fileout .eq. 'composite') fileout = 'composite.out'

   open(9,file=fileout)
   if (liketype .eq. 'S') then
     write(9,'(/,2A,///)') 'SINGLE-LOCUS ESTIMATES FOR: ',name
   else
     write(9,'(/,2A,///)') 'COMPOSITE LIKELIHOOD FOR:  ',name
   endif
   if (model .eq. 'M') then
     write(9,'(A,/)') 'POPULATION MODEL:  Moran/Coalescent'
   else
     write(9,'(A,/)') 'POPULATION MODEL:  Branching Process'
   endif
   if (approx .eq. '2') then
     write(9,'(A,/)') 'APPROXIMATION:     Second Order'
   else
     write(9,'(A)') 'APPROXIMATION:     First Order'
   endif
   if (liketype .eq. 'C') then
     write(9,'(//,A)') 'SPECIFIED MAP (Intermarker distances
1 given in kilobases)'
     write(9,'(A)') '------------------------------------
1--------------------'
     do i = 1, nummark
       write(9,'(9x,A)') markname(i)
       if (i .ne. nummark) write(9,'(f8.1)') kb(i+1)-kb(i)
     enddo
     write(9,'(//,A)') '    AGE       kb             Composite'
     write(9,'(A,/)')  '--------------------------------------'
   endif

   do age = agelim(1), agelim(2), agelim(3)

     if (model .eq. 'M') then
       lambda = dlog(dpopsize) / age
     else
       lambda = dpopsize**(1.0d0/age) - 1.0d0
     endif
     if (liketype .eq. 'S') then
       write(9,'(/,A)')
1            'Estimates given in kilobases.'
       write(9,'(////,A,f7.1,A,//)')
1 'Age assumed to be ',age,' generations.'
       write(9,'(A)')
1'MAP                     Estimate    Lower Limit    Upper Limit'
     write(9,'(A,/)')
```

```
1'-------------------------------------------------------------'

       if (kboff .eq. 0.0d0) kboff = 1.0d3
       do i = 1, nummark
         open(19,file='cABC0001.TMP')
         do dloc = kbmesh+1.0d-4, kboff+1.0d-4, kbmesh
           r = 0.50d0 * (1.0d0 - dexp(-2.0d-5 * dloc))
           llike = foa(maxmark, nummark, maxall, numall, model,
1                      approx, hessian, lambda, EXXt, EPd, hE,
2                      i, nd, ranks, pn, smooth, coval, age, r,
3                      mean, mutmark, dismut, mt, mt_1, SVE, Rt)
           write(19,'(f7.2,f9.6,f35.5)') dloc, r, llike
         enddo
         rewind(19)
         maxlike = -1.0d100
         do temp = kbmesh+1.0d-4, kboff+1.0d-4, kbmesh
           read(19,'(f7.2,f9.6,f35.5)') dloc, r, llike
           if (llike .ge. maxlike) then
             dlhat = dloc
             rhat = r
             maxlike = llike
           endif
         enddo
         rewind(19)
         ll = 0.0d0
         ul = kboff + 0.00010d0
         temp2 = 0.0d0
         do temp = kbmesh+0.00010d0, kboff+0.00010d0, kbmesh
           read(19,'(f7.2,f9.6,f35.5)') dloc, r, llike
           if (dloc .lt. dlhat) then
             if (llike .lt. maxlike - conf) ll = dloc
           else
             if (llike .lt. maxlike - conf) temp2=temp2+1.0d0
             if ((temp2 .gt. 0.90d0) .and. (temp2 .lt. 1.10d0))
1                                     ul = dloc
           endif
         enddo
         write(9,'(A,2x,f14.3,2f15.3)') markname(i), dlhat, ll, ul
         if (i .ne. nummark) write(9,'(/,8x,f7.1,A,/)')
1                                  kb(i+1)-kb(i), ' kb'
         close(19,status='delete')
       enddo
       write(9,'(//,A,f9.3,A)')
1 'NOTE:  If the upper confidence limit is equal to ',
2  kboff+1.0d-4, ','
       write(9,'(7x,A)')
1 'then the chosen off-end distance was too small.'
     else
       do dloc = 1.0d-4 - kboff, kb(nummark)+kboff+1.0d-4, kbmesh
         llike = 0.0d0
```

```
             do i = 1, nummark
               r = 0.50d0 *
    1              (1.0d0 - dexp(-0.000020d0*dabs(dloc-kb(i))))
               llike = llike + foa(maxmark, nummark, maxall, numall,
    1                 model, approx, hessian, lambda, EXXt, EPd, hE,
    2                      i, nd, ranks, pn, smooth, coval, age, r,
    3                      mean, mutmark, dismut, mt, mt_1, SVE, Rt)
             enddo
             write(9,'(f7.1,3x,f10.3,3x,f15.3)') age, dloc, llike
           enddo
         endif
       enddo
       close(9)
       goto 9999

9999   continue
       end
```

```
      subroutine menu(model, approx, liketype, kbmesh, kboff,
     1                agelim, dpopsize, smooth, coval, filein,
     1                fileout, input, run)
c
c       This subroutine is simply the main menu for the composite likelihood
c       program.  It allows the user to set and change parameters before
c       running the program.
c
c===============================================================================
c
c       Variable declarations
c
        implicit none

        character*1     model   !Population model to be used
                                !  B = Branching Process, M = Moran/Coalescent
        character*32    popmod  !text describing population model
        character*1     approx  !Type of approximation to the likelihood
                                !  1 = FOA,  2 = SOA
        character*13    atype   !text describing approximation type
        character*1     liketype        !Type of likelihood to form
                                        !  S = Single Marker,  C = Composite
        character*13    ltype           !text describing likelihood type
        real*8          kbmesh          !grid coarseness for the likelihood
        real*8          kboff           !distance in kb to calculate the
                                        !  likelihood from the ends of the map
        real*8          agelim(3)       !contains elements for forming age grid
        real*8          dpopsize        !current size of disease population
        character*1     smooth          !smooth  'Y' or 'N'
        real*8          coval           !cutoff falue for smoothing
        character*20    filein          !name of the input file
        character*20    fileout         !name of the output file
        character*1     input           !when input = I, input the data
        character*1     run             !when run = R, run the program
        logical*1       correct         !limits inputs to menu choices
        character*1     choice          !menu choice
        character*1     ch_vals(18)     !acceptable choices
        integer         i               !loop counter

        data ch_vals, popmod, atype, ltype
     1         / 'A','a','B','b','C','c','D','d','E','e','I','i',
     2           'R','r','Q','q','F','f',
     3           'Galton-Watson Branching Process','First Order',
     4           'Composite' /
c===============================================================================
c
c       Begin Execution
c
        run = 'q'
```

```
   do while ((run .ne. 'R') .and. (run .ne. 'r') .and.
1            (run .ne. 'Q'))
     correct = .false.
     do while (.not. correct)
       write(*,'(//////////////////////////////)')
       write(*,'(//////////////////////////////)')
       write(*,'(/,A,A)')
1            '=======================================',
2            '======================================='
       write(*,'(5x,a,5x,9x,A)')
1            'A - Population Model',popmod
       write(*,'(/,5x,a,5x,7x,A)')
1            'B - Approximation Type',atype
       write(*,'(/,5x,a,5x,10x,A)')
1            'C - Likelihood Type',ltype
       write(*,'(/,5x,a,5x,A,f6.1,A,f6.1,A,f6.1,A)')
1            'D - Parameters:',
2             'age:  from ',agelim(1),' to ',
3              agelim(2),' by ',agelim(3), ' generations'
       write(*,'(25x,A,2x,f12.0)')'# disease chromosomes:',dpopsize
       write(*,'(25x,A,13x,f5.1)')
1             'mesh size in kb:   ',kbmesh
       write(*,'(25x,A,13x,f7.1)')
1              'kb outside map:  ',kboff
       if (smooth .eq. 'Y') then
         write(*,'(/,5x,A,21x,A,A,10x,A,f5.2)')
1            'E - Smoothing',smooth,'es','cut-off value: ',coval
       else
         write(*,'(/,5x,A,21x,A,A)')
1            'E - Smoothing',smooth,'o'
       endif
       write(*,'(/,5x,A,12x,A,A)')
1            'F - Files:','Input:     ',filein
       write(*,'(27x,A,A)') 'Output:     ',fileout
       write(*,'(/,5x,18a,5x,A)')
1            'I - Input Data'
       write(*,'(/,5x,18a,5x,A)')
1            'R - Run'
       write(*,'(/,5x,A)')
1            'Q - Quit without running program'
       write(*,'(A,A)')
1            '=======================================',
2            '======================================='
       write(*,'(A)') 'Enter a letter to make modifications.'
       read(*,'(A)') choice
       do i = 1, 18
         if (choice .eq. ch_vals(i)) correct = .true.
       enddo
     enddo
     if ((choice .eq. 'a') .or. (choice .eq. 'A')) then
```

```
      if (model .eq. 'B') then
        model = 'M'
        popmod = 'Moran/Coalescent'
      else
        model = 'B'
        popmod = 'Galton-Watson Branching Process'
      endif
     endif
     if ((choice .eq. 'b') .or. (choice .eq. 'B')) then
      if (approx .eq. '1') then
        approx = '2'
        atype  = 'Second Order'
      else
        approx = '1'
        atype = 'First Order'
      endif
     endif
     if ((choice .eq. 'c') .or. (choice .eq. 'C')) then
      if (liketype .eq. 'C') then
        liketype = 'S'
        ltype  = 'Single Marker'
      else
        liketype = 'C'
        ltype = 'Composite'
      endif
     endif
     if ((choice .eq. 'd') .or. (choice .eq. 'D')) then
      write(*,'(////,7x,A,////)')
  1' ***  These parameters are also read from the input file.  ***'
      write(*,'(A)') 'NOTE:  Ages must be given in units of
  1 generations.'
100   write(*,'(//,A,/)') 'Enter the smallest age of the disease
  1 to consider.'
      read(*,*,err=100) agelim(1)
101   write(*,'(/,A,/)') 'Enter the largest age of the disease
  1 to consider.'
      read(*,*,err=101) agelim(2)
102   write(*,'(/,A,/)') 'Enter the mesh size for disease age.'
      read(*,*,err=102) agelim(3)
103   write(*,'(/,A,/)') 'Enter the number of disease
  1 chromsomes in the population.'
        read(*,*,err=103) dpopsize
104   write(*,'(/,A,/)') 'Enter the mesh size, in kb, for
  1 the grid of likelihood evaluations.'
      read(*,*,err=104) kbmesh
105   write(*,'(/,A)') 'Enter the distance, in kb, where
  1 the likelihood is to be evaluated'
      write(*,'(A,/)') 'beyond the ends of the map.'
      read(*,*,err=105) kboff
     endif
     if ((choice .eq. 'e') .or. (choice .eq. 'E')) then
       if (smooth .eq. 'Y') then
         smooth = 'N'
       else
         smooth = 'Y'
         write(*,'(/,2A)') 'Enter the inverse smoothing parameter',
1         ' (larger values imply less smoothing).'
         read(*,*) coval
         if (coval .lt. 0.0d0) smooth = 'N'
       endif
     endif
     if ((choice .eq. 'f') .or. (choice .eq. 'F')) then
       input = 'q'
       write(*,'(/,A,/)') 'Enter the name of the file containing
1 the data.'
       read(*,*) filein
       write(*,'(/,A,/)') 'Enter the name of the file to
1 contain the output.'
       read(*,*) fileout
     endif
     if ((input .ne. 'I') .and. ((choice .eq. 'i') .or.
1                                (choice .eq. 'I'))) then
       input = 'I'
       run = 'r'
     endif
     if ((choice .eq. 'r') .or. (choice .eq. 'R')) then
       if (input .ne. 'd') then
         input = 'q'
         write(*,'(////,A,//,A,////)')
1     '              *** Input data before running program! *** ',
2     '                      Press <RETURN> to continue.'
         read(*,*)
       else
         input = 'I'
         run = 'R'
       endif
     endif
     if ((choice .eq. 'q') .or. (choice .eq. 'Q')) run = 'Q'
   enddo

   return
   end
```

```
      subroutine datain(maxmark,maxall,filein,name,dpopsize,dismut,
     1                  nummark, markname, numall, nd, nn, kb, pd,
     2                  pn, mutmark, agelim, kboff, kbmesh, input)

      implicit none

      integer maxmark                 !maximum number of markers
      integer maxall                  !maximum number of alleles at each mark
      character*20 filein             !name of the input file
      character*40 name               !name of the disease
      real*8 dpopsize                 !current size of disease population
      real*8 dismut                   !disease mutation rate
      integer nummark                 !number of markers
      character*16 markname(maxmark)  !names of markers
      integer numall(maxmark)         !number of alleles at each marker
      real*8 nd(maxmark,maxall)       !sampled disease chromosomes
      real*8 nn(maxmark,maxall)       !sampled normal chromosomes
      real*8 kb(maxmark)              !kilobases between adjacent markers
      real*8 pd(maxmark,maxall)       !disease marker allele frequencies
      real*8 pn(maxmark,maxall)       !normal marker allele frequencies
      real*8 mutmark(maxmark,maxall,maxall)  !marker mutation matrices
      real*8 totd, totn               !counts the total sample sizes
      real*8 agelim(3)                !years to consider
      real*8 kboff                    !distance in kb to calculate the
                                      !  likelihood from the ends of the map
      real*8 kbmesh                   !grid coarseness for the likelihood
      character*1 input               !if input=I, then ok to input data
      integer i, j, k                 !loop counters

      open(1,file=filein,status='old',err=10)

      read(1,'(A)') name
      read(1,*) dpopsize
      read(1,*) dismut
      read(1,*)
      read(1,*) nummark

      kb(1) = 0.0d0
      do i = 1, nummark
        read(1,*)
        read(1,'(A)') markname(i)
        read(1,*) numall(i)
        read(1,*) (nd(i,j), j = 1, numall(i))
        read(1,*) (nn(i,j), j = 1, numall(i))
        do j = 1, numall(i)
          read(1,*) (mutmark(i,j,k),k=1,numall(i))
        enddo
        if (i .ne. nummark) then
          read(1,*) kb(i+1)
          kb(i+1) = kb(i+1) + kb(i)
          endif

          totd = 0.0d0
          totn = 0.0d0
          do j = 1, numall(i)
            totd = totd + nd(i,j)
            totn = totn + nn(i,j)
          enddo
          do j = 1, numall(i)
            pd(i,j) = nd(i,j)/totd
            pn(i,j) = nn(i,j)/totn
          enddo

        enddo

        read(1,*)
        read(1,*) (agelim(j), j=1,3)
        read(1,*)
        read(1,*) kboff
        read(1,*)
        read(1,*) kbmesh

        close(1)
        input = 'd'
        goto 11
10      write(*,'(/////,A,/)')
     1  '                  *** The input file does not exist! ***'
        write(*,'(/,A,////)')
     1  '                        Press <RETURN> to continue.'
        read(*,*)
        input = 'q'
11      continue
        return
        end
        subroutine rankeach(maxmark,nummark,maxall,numall,nd,ranks)
c
c       This subroutine ranks the observed disease chromosome counts
c       of each marker.
c
c==============================================================================
c
c       Variable declarations
c
        implicit none

        integer         maxmark         !the maximum number of markers
        integer         nummark         !the actual number of markers
        integer         maxall          !the maximum number of alleles
        integer         numall(maxmark) !number of alleles at each marker
        real*8  nd(maxmark,maxall)      !sampled disease chromosomes
        integer ranks(maxmark,maxall)   !ranks of observed disease chromosome
                                        !  counts for each marker
        integer         m,i             !loop counters
        integer         ip1             !equals i + 1
        integer         istop           !controls stopping on one pass of sort
        logical*1       sorted          !is the vector sorted?
        integer         temp            !used to exchange values in rank

c==============================================================================
c
c       Begin execution
c

c  For each marker,
        do m = 1, nummark

c  Initialize the order array
          do i = 1, numall(m)
            ranks(m,i) = i
          enddo

c  Sort the rank array
          istop = numall(m) - 1
          sorted = .false.
          do while (.not. sorted)
            sorted = .true.
            do i = 1, istop
              ip1 = i+1
              if (nd(m,ranks(m,i)) .lt. nd(m,ranks(m,ip1))) then
                temp = ranks(m,i)
                ranks(m,i) = ranks(m,ip1)
                ranks(m,ip1) = temp
                sorted = .false.
              endif
```

```
    enddo
    istop = istop-1
  enddo

enddo

return
end
```

```
        real*8 function foa(maxmark, nummark, maxall, numall, model,
     1                      approx, hessian, lambda, EXXt, EPd, hE,
     2                      m, nd, ranks, pn, smooth, coval, age, r,
     3                      mean, mutmark, dismut, mt, mt_1, SVE, Rt)
c
c       This function calculates the first order approximation to the
c       likelihood, based on the first moments of either a Galton-
c       Watson branching process or a Moran/Coalescent model.
c
c       It assumes that the first element in nd (and pn) represents the
c       ancestral allele.
c
c===============================================================================
c
c       Variable Declarations
c
        implicit none

        integer         maxmark         !the maximum number of markers
        integer         nummark
        integer         maxall          !the maximum number of alleles
        integer         numall(maxmark) !the observed number of alleles
        character*1     model           !which model to use
                                        !  B = branching process
                                        !  M = Moran/Coalescent
        character*1     approx  !Type of approximation to the likelihood
                                !  1 = FOA,  2 = SOA
        real*8   hessian(maxall,maxall) !the Hessian of prod(pid^nid)
        real*8   lambda                 !the growth parameter
        real*8   EXXt(maxall,maxall)    !the matrix of second moments for
                                        !   the counts
        real*8   EPd(maxall)            !the expected disease allele freqs
                                        !  (need to be in sorted order)
        real*8  hE(maxall,maxall)       !hessian * EXXt

        integer         m               !which marker
        real*8   nd(maxmark,maxall)     !the disease allele counts
        integer  ranks(maxmark,maxall)  !the ranks of the disease allele counts
        real*8   pn(maxmark,maxall)     !the normal allele frequencies
        character*1     smooth          !will we smooth the log likelihood?
        real*8          coval           !the cut-off to be used in smoothing
        real*8          age             !the assumed age
        real*8          r               !the assumed recombination coefficient
        real*8  mean(maxall,maxall)     !one-step mean matrix
        real*8  mutmark(maxmark,maxall,maxall)  !mutation matrices
        real*8  dismut                  !disease mutation matrix
        real*8  mt(maxall,maxall)       !mean^age (matrix)
        real*8  mt_1(maxall,maxall)     !more powers of mean
        real*8  SVE(maxall,maxall)      !sum of V_i*EX_i^(t-1)
        real*8  Rt(maxall,maxall)       !the joint allele probabilities
        real*8          soab            !function that calculates the second-
                                        !  order correction

        real*8          p1d             !most common disease allele frequency
        real*8          rhat            !the mle
        real*8          llmax           !the max value of the log likelihood
        real*8          ll              !temporary variable for calculating the
                                        !  log-likelihood value
        real*8          decay           !decay of disequilibrium
        integer         i, j            !loop counters
        real*8  dpopsize                !size of the disease population
        real*8  rtint                   !function that calculates the integrand
                                        !  for Rt
        external        rtint

c==============================================================================
c
c       Begin execution
c

        if (model .eq. ’M’) then        !Moran/Coalescent model, get intensity
          do i = 1, numall(m)           !  matrix
            do j = 1, numall(m)
              mean(i,j) = (1-r)*mutmark(m,ranks(m,i),ranks(m,j)) +
     1                    (r+dismut) * pn(m,ranks(m,j))
            enddo
            mean(i,i) = mean(i,i) - 1.0d0 - dismut
          enddo

          dpopsize = dexp(lambda*age)

c  calculate the expected frequencies

c  calculate the exponential of the intensity matrix and take the first row
c    as the expected allele freqs.
          call expmat(maxall,numall(m),age,mean,hE,EXXt,mt,EPd)

        else            !branching process model, calculate moment matrix
          do i = 1, numall(m)
            do j = 1, numall(m)
              mean(i,j) = (1+lambda) *
     1                    ((1-r)*mutmark(m,ranks(m,i),ranks(m,j)) +
     2                     (r+dismut) * pn(m,ranks(m,j)))
            enddo
          enddo

c  calculate the expected frequencies (approximations)

c  first, calculate the age power of the matrix
```

```
          call matpow(maxall, numall(m), age, mean, hE, mt)

c  then approximate expected allele frequencies
          ll = 0.0d0
          do i = 1, numall(m)
            ll = mt(1,i) + ll
          enddo
          do i = 1, numall(m)
            EPd(i) = mt(1,i) / ll
          enddo
        endif

c  Finally, calculate foa to the log likelihood (for either bp or M/c)
        ll = 0.0d0
        do i = 1, numall(m)
          ll = ll + nd(m,ranks(m,i)) * dlog(EPd(i))
        enddo

c  If we want to calculate the second-order approximation, we need to
c  calculate the correction.
        if (approx .eq. '2') then

c  calculate the Hessian for the log-likelihood
          do i = 1, numall(m)
            hessian(i,i) = -nd(m,ranks(m,i)) / (EPd(i)**2)
            do j = i+1, numall(m)
              hessian(i,j) = 0.0d0
              hessian(j,i) = 0.0d0
            enddo
          enddo

c  calculate the second-order approximation
          if (model .eq. 'M') then
c  use the Moran/Coalescent SOA
c  get the initial stage of Rt (as indicated when writing the expression with
c    initial conditions separated)
            p1d = dexp((dexp(-lambda*age)-1.0d0)/lambda)  !a temporary constant
            do i = 1, numall(m)
              do j = i, numall(m)
                Rt(i,j) = EPd(i) * EPd(j) * p1d
                Rt(j,i) = Rt(i,j)
              enddo
            enddo
c  use rhat as a temporary variable containing the integral, and update the
c    initial Rt to get the final version
            p1d = p1d * dexp(1.0d0/lambda) * dexp(-lambda*age)  !temp. constant
            do i = 1, numall(m)
              do j = i, numall(m)
c  Here, we use EXXt, SVE, EPd and mt_1 as temporary matrices used in the
c    evaluation of the integral
```

```
                call trapzoid(maxall, numall(m), i, j, lambda, mean,
     1                        mt, EXXt, SVE, mt_1, EPd, 0.0d0,
     2                        age, rhat)
                Rt(i,j) = Rt(i,j) + p1d * rhat
                Rt(j,i) = Rt(i,j)
              enddo
            enddo
c  Turn the Rt values into noncentral moments
            do i = 1, numall(m)
              Rt(i,i) = Rt(i,i) + (mt(1,i) - Rt(i,i)) / dpopsize
              do j = i+1, numall(m)
                Rt(i,j) = Rt(i,j) - Rt(i,j)/dpopsize
                Rt(j,i) = Rt(i,j)
              enddo
            enddo
c  We have a hessian, and the matrix of noncentral second moments,
c  so we can calculate the correction to the log likelihood
            do i = 1, numall(m)
              do j = 1, numall(m)
                hE(i,j) = hessian(i,i) * Rt(i,j)
              enddo
            enddo
c  Use p1d as the variable containing the correction
            p1d = 0.0d0
            do i = 1, numall(m)
              p1d = p1d + hE(i,i)               !trace part of correction
              p1d = p1d - hessian(i,i) * mt(1,i)**2     !other part of correct.
            enddo
            ll = ll + p1d / 2.0d0
          else
c  use the Galton-Watson branching process SOA
            ll = ll + soab(maxmark, nummark, maxall, numall,
     1                     m, ranks, pn, age, r, hessian,
     2                     lambda, EXXt, EPd, hE,
     3                     mean, mt, mt_1, SVE)
          endif
        endif

        if (smooth .eq. 'Y') then
          p1d = 0.0d0
          do i = 1, numall(m)
            p1d = p1d + nd(m,ranks(m,i))
          enddo
          p1d = nd(m,ranks(m,1)) / p1d

c  Get the mle (first order)
          if (model .eq. 'M') then
            rhat = -1.0d0/age * dlog((p1d - pn(m,ranks(m,1))) /
     1                            (1.0d0 - pn(m,ranks(m,1))))
            decay = dexp(-rhat*age)
     else
       rhat = 1.0d0 - ((p1d - pn(m,ranks(m,1))) /
1                   (1.0d0 - pn(m,ranks(m,1))))**(1.0d0/age)
       decay = (1.0d0 - rhat)**age
     endif
     llmax = nd(m,ranks(m,1)) *
1       dlog(pn(m,ranks(m,1)) + (1.0d0-pn(m,ranks(m,1)))*decay)
     do i = 2, numall(m)
       llmax = llmax + nd(m,ranks(m,i)) *
1              dlog(pn(m,ranks(m,i)) * (1.0d0 - decay))
     enddo
     if (r .lt. rhat) ll = max(ll,llmax-coval)
   endif

   foa = ll

   return
   end
```

```
      real*8 function soab(maxmark, nummark, maxall, numall,
     1                     m, ranks, pn, age, r, hessian,
     2                      lambda, EXXt, EPd, hE,
     3                      mean, mt, mt_1, SVE)
c
c       This function calculates the correction that gives the
c       second order approximation to the likelihood.
c
c       This function is appropriate only for the branching process
c       model.
c
c==============================================================================
c
c       Variable declarations
c
        implicit none

        integer         maxmark         !the maximum number of markers
        integer         nummark
        integer         maxall          !the maximum number of alleles
        integer         numall(maxmark) !the observed number of alleles
        integer         m               !which marker
        integer  ranks(maxmark,maxall)  !the ranks of the disease allele counts
        real*8   pn(maxmark,maxall)     !the normal allele frequencies
        real*8          age             !the assumed age
        real*8          r               !the assumed recombination coefficient
        real*8   hessian(maxall,maxall) !the Hessian of prod(pid^nid)
        real*8   lambda                 !the growth parameter
        real*8   EXXt(maxall,maxall)    !the matrix of second moments for
                                        !   the counts
        real*8   EPd(maxall)            !the expected disease allele freqs
                                        !  (need to be in sorted order)
        real*8  hE(maxall,maxall)       !hessian * EXXt
        real*8  mean(maxall,maxall)     !one-step matrix of first moments
        real*8  mt(maxall,maxall)       !powers of mean
        real*8  mt_1(maxall,maxall)     !more powers of mean
        real*8  SVE(maxall,maxall)      !sum of V_i*EX_i^(t-1)

        real*8  t                       !age loop counter
        integer         i, j            !loop counters
        real*8  pop2                    !(1+lambda)^(2t)
        real*8  correct                 !used to calculate the soa correction

c==============================================================================
c
c       Begin execution
c
c

        pop2 = (1.0d0 + lambda)**(2*age)
```

```
c  Calculate the matrix of noncentral second moments for the counts

c  get the age-th power of the moment matrix and
c    and get the first part of EXXt (the squared first moment)
        call matpow(maxall, numall(m), age, mean, hE, mt)
        do i = 1, numall(m)
          do j = 1, numall(m)
            EXXt(i,j) = mt(1,i) * mt(1,j)
          enddo
        enddo

c  get the rest of EXXt
        do t = 1.0d0, age, 1.0d0
c  calculate (age-t)-th power of the mean matrix
          call matpow(maxall, numall(m), age-t, mean, hE, mt)

c  get the sum of the Vi*Ei,j-1  (this equals diag[mean**t])
          do i = 1, numall(m)
            do j = i+1, numall(m)
              SVE(i,j) = 0.0d0
              SVE(j,i) = 0.0d0
            enddo
          enddo

          call matpow(maxall, numall(m), t, mean, hE, mt_1)
          do i = 1, numall(m)
            SVE(i,i) = mt_1(1,i)
          enddo

c  mt contains mean^(age-t), now we calculate mean'^(age-t)*sve (matrix mult)
          do i = 1, numall(m)
            do j = 1, numall(m)
              hE(i,j) = mt(j,i) * SVE(j,j)      !use mt(j,i) since we want to
                                                !  use mt' here
                                                !this works since SVE is diag.
            enddo
          enddo

c  now hE contains mean'^(age-t)*sve and mt contains mean^(age-t), we can
c    finally get mean'^(age-t)*sve*mean^(age-t) (we store it in SVE)
          call matmult(maxall, numall(m), hE, mt, SVE)

c  we now update EXXt
          do i = 1, numall(m)
            do j = 1, numall(m)
              EXXt(i,j) = EXXt(i,j) + SVE(i,j)
            enddo
          enddo
        enddo
```

```
c  We have a hessian, and the matrix of noncentral second moments,
c  so we can calculate the correction to the log likelihood
        do i = 1, numall(m)
          do j = 1, numall(m)
            hE(i,j) = hessian(i,i) * EXXt(i,j)
          enddo
        enddo

        correct = 0.0d0

        do i = 1, numall(m)
          correct = correct + hE(i,i)/(pop2)    !get trace part of correction,
                                                !  scaling it for counts
          correct = correct-EPd(i)**2*hessian(i,i)      !get last part of
        enddo                                           !  correction

c  Correction for second-order approximation to the log-likelihood
        soab = (correct / 2.0d0)

        return
        end
```

```
      subroutine matmult(maxall, numall, A, B, C)
c
c     This routine calculates the matrix product of A times B, and returns
c     it in the matrix C (this is written for square matrices only).
c
      integer maxall                  !the maximum number of alleles allowed
      integer numall                  !the number of marker alleles
      real*8  A(maxall,maxall)        !the first matrix
      real*8  B(maxall,maxall)        !the second matrix
      real*8  C(maxall,maxall)        !the output of AB

      do i = 1, numall
        do j = 1, numall
          C(i,j) = 0.0d0
          do k = 1, numall
            C(i,j) = C(i,j) + A(i,k) * B(k,j)
          enddo
        enddo
      enddo

      return
      end
```

```
        subroutine matpow(m, n, pow, A, B, C)
c
c       This routine calculates the matrix power of A^pow, and returns
c       it in the matrix C.
c
        integer m               !the maximum number of alleles allowed
        integer n               !the number of marker alleles
        real*8  pow             !power of matrix to be calculated
        real*8  A(m,m)          !the matrix
        real*8  B(m,m)          !a temporary matrix (same size as A)
        real*8  C(m,m)          !the output of A^pow

        real*8  t               !counting variable

        do i = 1, n
          do j = 1, n
            B(i,j) = A(i,j) !copy the contents of A into B
            C(i,j) = A(i,j) !copy the contents of A into C (needed if pow=1)
          enddo
        enddo

        do t = 2.0d0, pow, 1.0d0
          call matmult(m, n, A, B, C)   !multiply A * B (B = A^(t-1)) => C=A^t
          if (t .lt. pow) then
            do i = 1, n
              do j = 1, n
                B(i,j) = C(i,j)         !set B = A^t
              enddo
            enddo
          endif
        enddo

        return
        end
        subroutine expmat(m, n, t, A, B, C, D, E)
c
c       This routine calculates an approximation to the exponential
c       of the matrix A, and returns the result in the matrix C.
c
        integer m               !the maximum number of alleles allowed
        integer n               !the number of marker alleles
        real*8  t               !age of the disease mutation  (e^(At))
        real*8  A(m,m)          !the matrix
        real*8  B(m,m)          !a temporary matrix (same size as A)
        real*8  C(m,m)          !a temporary matrix (same size as A)
        real*8  D(m,m)          !the output (exp(At))
        real*8  E(m)            !the first row of D (expected allele freqs)

        real*8  error           !error of the approximation
        real*8  iter            !iterates the approximation
        real*8  fact            !function for calculating factorials
        real*8  tfact           !temp variable so only calculate fact once

c  Do some initializations
        error = 1.0d30
        do i = 1, n
          do j = 1, n
              A(i,j) = A(i,j) * t       !scale A by age of disease
              D(i,j) = A(i,j)
          enddo
          D(i,i) = D(i,i) + 1.0d0       !D now contains the first order
                                        !  approx to the exponential matrix
                                        !  I + A*t
        enddo

c  Now move on to compute a higher-order approx to the exponential matrix
        iter = 2.0d0
        do while (error .gt. 1.0d-20)
          error = 0.0d0
          call matpow(m, n, iter, A, B, C)  !C contains (A*t)^iter

          tfact = fact(iter)
          do i = 1, n
            do j = 1, n
              D(i,j) = D(i,j) + C(i,j)/tfact
              if (i .eq. 1) then
                error = (E(j) - D(i,j))**2 + error
                E(j) = D(i,j)
              endif
            enddo
          enddo
          iter = iter + 1.0d0
        enddo
```

```
do i = 1, n
  do j = 1, n
    A(i,j) = A(i,j) / t
  enddo
enddo

return
end
```

```
        real*8 function fact(num)

c
c       This function returns the factorial of NUM.
c
c       NOTE:  This is not to be used for large values of NUM!!!
c

        real*8  num             !the number
        real*8  count           !loop counter
        real*8  temp            !temporary variable

        temp = 1.0d0
        do count = 2.0d0, num, 1.0d0
          temp = temp * count
        enddo

        fact = temp

        return
        end
```

```
      real*8 function rtint(maxall, numall, row, col, lambda,
     1                      mean, mt, temp1, temp2, mx,
     2                      temp3, x)
c
c     This function returns the value of the integrand required for
c     the Moran second order approximation.
c
      integer maxall          !maximum number of alleles
      integer numall          !number of alleles
      integer row             !which row of Rt to calculate
      integer col             !which column of Rt to calculate
      real*8  lambda          !growth parameter
      real*8  mean(maxall,maxall)     !intensity matrix
      real*8  mt(maxall,maxall)       !t-generation transition matrix
      real*8  temp1(maxall,maxall)    !a temporary matrix
      real*8  temp2(maxall,maxall)    !a temporary matrix
      real*8  mx(maxall,maxall)       !x-generation transition matrix
      real*8  temp3(maxall)           !a temporary vector
      real*8  x               !point at which to evaluate the function

      call expmat(maxall, numall, x, mean, temp1, temp2, mx, temp3)

      do i = 1, numall
        do j = 1, numall
          temp1(i,j) = mt(j,i) * mx(1,j)      !P'(t) PI(t)
        enddo
      enddo

      call matmult(maxall, numall, temp1, mt, temp2)

      rtint = temp2(row,col) * dexp(-dexp(-lambda*x)/lambda)

      return
      end


      subroutine trapzoid(maxall, numall, row, col, lambda,
     1                    mean, mt, temp1, temp2, mx,
     2                    temp3, a, b, st)

      integer maxall          !maximum number of alleles
      integer numall          !number of alleles
      integer row             !which row of Rt to calculate
      integer col             !which column of Rt to calculate
      real*8  lambda          !growth parameter
      real*8  mean(maxall,maxall)     !intensity matrix
      real*8  mt(maxall,maxall)       !t-generation transition matrix
      real*8  temp1(maxall,maxall)    !a temporary matrix
      real*8  temp2(maxall,maxall)    !a temporary matrix
      real*8  mx(maxall,maxall)       !x-generation transition matrix
      real*8  temp3(maxall)           !a temporary vector
      real*8  a                       !lower limit of integration
      real*8  b                       !upper limit of integration
      real*8  st                      !sum of trapeziods (integral estimate)

      real*8  rtint                   !the function to evaluate
      external rtint

      real*8  n_traps                 !number of intervals
      real*8  x                       !where we evaluate the function
      real*8  y                       !functional value at x
      real*8  int_len                 !length of interval

      parameter (n_traps = 100)

      int_len = (b - a) / n_traps

      st = 0.0d0

      do x = a, b, int_len
        if (x .eq. 0.0d0) then
          y = mt(1,row)*mt(1,col)*dexp(-dexp(-lambda*x)/lambda)
        else
          y = rtint(maxall, numall, row, col, lambda,
     1              mean, mt, temp1, temp2, mx, temp3, x)
        endif
        if ((x .eq. a) .or. (x .eq. b)) then
          st = st + y
        else
          st = st + 2.0d0 * y
        endif
      enddo

      st = int_len * st / 2.0d0

      return

      end
```

Appendix B

Data Files

B.1 Cystic Fibrosis

In this section, we include the data file used in the composite likelihood program in Appendix A to generate the no-mutation composite log likelihood in Figure 7.17. The data were published by Kerem *et al.* [39].

```
Cystic Fibrosis                          !Name of the disease
2000000                                  !current number of disease chromosomes
0.000                                    !disease mutation rate

22              !number of markers
                !blank line  (repeat this sequence of lines for each marker)
metD BanI       !name of the marker
2               !number of marker alleles
48 25           !disease marker allele counts
28 59           !normal marker allele counts (in same order as disease ones)
1 0             !marker alleles x marker alleles mutation transition matrix
0 1
9               !distance in kb to next marker

metD TaqI
2
75 4
74 19
1 0
0 1
15.8

metH TaqI
2
49 20
45 38
1 0
0 1
500

E6
2
62 17
58 42
1 0
0 1
```

```
10

E7
2
57 16
51 40
1 0
0 1
20

pH131
2
47 33
18 81
1 0
0 1
15

W3D1.4
2
47 33
22 82
1 0
0 1
25

XV2C
2
53 11
39 37
1 0
0 1
20

HincII
2
69 7
31 56
1 0
0 1
20

BglII
2
69 9
27 62
1 0
0 1
20
```

```
KM19
2
70 10
30 69
1 0
0 1
30

E2.6
2
55 6
26 34
1 0
0 1
25

H2.8A
2
55 9
22 52
1 0
0 1
35

E4.1
2
64 8
38 37
1 0
0 1
35

J44
2
70 6
40 44
1 0
0 1
80

AccI
2
60 15
14 67
1 0
0 1
10

HaeIII
2
```

```
61 15
14 72
1 0
0 1
20

T6/20
2
66 8
21 56
1 0
0 1
10

H1.3
2
69 7
35 53
1 0
0 1
50

CE1.0
2
73 3
81 8
1 0
0 1
585

J3.11
2
38 36
36 62
1 0
0 1
100

J29
2
36 36
26 55
1 0                    !don't need distance to next marker (there isn't one)
0 1
                        !blank line
200 200 1               !generational limits to consider, plus mesh size

50                      !# of kilobases off the end of the map

1                       !mesh size for likelihood (in kb)
```

B.2 Diastrophic Dysplasia

In this section, we include the data file used in the composite likelihood program printed in Appendix A to generate the composite log likelihoods in Figure 7.19. The data were published by Hästbacka *et al.* [23].

```
Diastrophic Displasia           !Name of the disease
200000                          !number of disease chromosomes
0.00000                         !disease mutation rate

10                              !number of markers
               !blank line  (repeat this and the next lines for each marker)
D5S372         !name of the marker
2              !number of marker alleles
93 61          !disease marker allele counts
16 103         !normal marker allele counts (in same order as disease ones)
1 0            !marker alleles x marker alleles mutation transition matrix
0 1
775            !distance in kb to next marker

BT1
2
139 13
 5 117
1 0
0 1
45

CSF1R/EcoRI
2
150 8
12 116
1 0
0 1
35

CSF1R/TAGA
2
144 6
46 82
1 0
0 1
3

CSF1R/Sty1
2
151 7
```

```
34 93
1 0
0 1
3

CSF1R/CA
2
124 29
20 108
1 0
0 1
5

CSF1R/CCT
2
97 27
5 125
1 0
0 1
25

PDGFRB/BgII
2
94 47
36 87
1 0
0 1
30

PDGFRB/EcoRI
2
100 51
28 91
1 0
0 1
1000

RPS14
2
99 51
57 63
1 0
0 1

100 100 1               !generations to consider

0                       !kb off map end

1                       !kb mesh length
```

B.3 Huntington's Disease

In this section, we include the data file used in the composite likelihood program of Appendix A to generate the composite log likelihood in Figure 7.21. The data were published by MacDonald *et al.* [51].

```
Huntington's Disease                     !Name of the disease
100000                                   !current number of disease chromosomes
0.00000001                               !disease mutation rate

25              !number of markers
                !blank line  (repeat this and the next lines for each marker)
D4S111 PstI     !name of the marker
2               !number of marker alleles
17 26           !disease allele counts
36 61           !normal allele counts (in same order as disease alleles)
0.999 0.001     !marker alleles x marker alleles mutation transition matrix
0.001 0.999
5               !kb to next marker

D4S111 BelI
2
40 27
107 45
0.999 0.001
0.001 0.999
270

D4S115
2
18 29
27 84
0.999 0.001
0.001 0.999
20

D4S96
2
58 36
195 132
0.999 0.001
0.001 0.999
415

D4S168
2
29 28
```

```
68 58
0.999 0.001
0.001 0.999
105

D4S113
2
12 58
26 128
0.999 0.001
0.001 0.999
80

D4S186
2
57 12
132 24
0.999 0.001
0.001 0.999
55

D4S114
2
41 12
94 26
0.999 0.001
0.001 0.999
30

D4S98
2
38 129
38 352
0.999 0.001
0.001 0.999
400

D4S43 Sau96
2
50 6
111 14
0.999 0.001
0.001 0.999
5

D4S43 HincII
2
25 40
75 75
0.999 0.001
```

```
0.001 0.999
5

D4S43 StuI
2
29 38
47 108
0.999 0.001
0.001 0.999
180

D4S183
2
27 39
55 94
0.999 0.001
0.001 0.999
330

D4S182
2
40 29
66 81
0.999 0.001
0.001 0.999
190

D4S95 TaqI
2
53 94
155 285
0.999 0.001
0.001 0.999
5

D4S95 AccI
2
109 25
286 139
0.999 0.001
0.001 0.999
115

D4S127 Pvu II
2
57 16
97 67
0.999 0.001
0.001 0.999
5
```

```
D4S127 Stu I
2
18 53
60 99
0.999 0.001
0.001 0.999
330

D4S180 BamHI
2
25 46
28 124
0.999 0.001
0.001 0.999
5

D4S180 XmnI
2
24 20
57 51
0.999 0.001
0.001 0.999
150

D4S125
2
35 26
93 40
0.999 0.001
0.001 0.999
250

D4S126
2
34 34
61 99
0.999 0.001
0.001 0.999
205

D4S10 HindIII
2
49 22
275 109
0.999 0.001
0.001 0.999
5

D4S10 EcoRI
```

```
2
39 48
206 220
0.999 0.001
0.001 0.999
5

D4S10 BgII
2
27 17
108 60
0.999 0.001
0.001 0.999

200 200 1               !generational limits to consider, plus mesh size

0                       !# of kilobases off the end of the map

5                       !mesh size for likelihood (in kb)
```

Appendix C

Simulation Results

This appendix contains the results of the simulation performed in Chapter 7. The first section contains multivariate and univariate analysis of variance tables for each of the four possible settings of the location of the disease mutation relative to the first marker locus. It also contains estimates of the logarithm of the absolute error (plus one), tabulated at the levels of the interactions that were significant at the $\alpha = 0.01$ level.

The second section contains maximum likelihood analysis of variance tables for the coverage probabilities. It also contains estimates of the coverage probablities, tabluated at the levels of the interactions that were significant at the $\alpha = 0.01$ level.

Note that the names of the factors are indicated in Table 7.7.

C.1 Prediction Error

C.1.1 Analysis of Variance Tables

```
KB_D=-50

MULTIVARIATE ANALYSIS OF VARIANCE

Source           S    M     N   Wilk's Lambda      F      df1       df2  Pr > F

A1               1   0.5   329    0.88181849    29.4844     3       660  0.0001
A2               1   0.5   329    0.81352990    50.4264     3       660  0.0001
A1*A2            1   0.5   329    0.96565005     7.8258     3       660  0.0001
DSAMP            2    0    329    0.98241743     1.9600     6      1320  0.0684
A1*DSAMP         2    0    329    0.99040473     1.0631     6      1320  0.3827
A2*DSAMP         2    0    329    0.97971913     2.2654     6      1320  0.0352
A1*A2*DSAMP      2    0    329    0.98461972     1.7116     6      1320  0.1147
KB_B             2    0    329    0.77816608    29.3943     6      1320  0.0001
A1*KB_B          2    0    329    0.88204275    14.2491     6      1320  0.0001
A2*KB_B          2    0    329    0.88373650    14.0245     6      1320  0.0001
A1*A2*KB_B       2    0    329    0.88169760    14.2949     6      1320  0.0001
DSAMP*KB_B       3    0    329    0.98407946     0.8855    12  1746.487  0.5614
A1*DSAMP*KB_B    3    0    329    0.99079094     2.0448     3       660  0.1063
A1*M1            1   0.5   329    0.99315374     1.5166     3       660  0.2090
A2*M1            1   0.5   329    0.99651976     0.7683     3       660  0.5120
A1*A2*M1         1   0.5   329    0.99681205     0.7036     3       660  0.5501
DSAMP*M1         2    0    329    0.99044071     1.0591     6      1320  0.3852
```

```
A1*DSAMP*M1      2    0     329    0.99351547    0.7168     6       1320  0.6361
A2*DSAMP*M1      2    0     329    0.99151982    0.9388     6       1320  0.4660
KB_B*M1          2    0     329    0.98081308    2.1414     6       1320  0.0462
A1*KB_B*M1       2    0     329    0.97833148    2.4230     6       1320  0.0247
A2*KB_B*M1       2    0     329    0.99256415    0.8225     6       1320  0.5524
DSAMP*KB_B*M1    3    0     329    0.98771679    0.6815    12   1746.487  0.7708
M2               1    0.5   329    0.97974983    4.5471     3        660  0.0036
A1*M2            1    0.5   329    0.99898621    0.2233     3        660  0.8802
A2*M2            1    0.5   329    0.96816043    7.2351     3        660  0.0001
A1*A2*M2         1    0.5   329    0.99296082    1.5596     3        660  0.1980
DSAMP*M2         2    0     329    0.97273943    3.0614     6       1320  0.0056
A1*DSAMP*M2      2    0     329    0.97934240    2.3082     6       1320  0.0320
A2*DSAMP*M2      2    0     329    0.98127429    2.0892     6       1320  0.0518
KB_B*M2          2    0     329    0.96160128    4.3495     6       1320  0.0002
A1*KB_B*M2       2    0     329    0.99499948    0.5521     6       1320  0.7686
A2*KB_B*M2       2    0     329    0.97517755    2.7824     6       1320  0.0108
DSAMP*KB_B*M2    3    0     329    0.97095141    1.6307    12   1746.487  0.0768
M1*M2            1    0.5   329    0.99324374    1.4965     3        660  0.2143
A1*M1*M2         1    0.5   329    0.99606476    0.8692     3        660  0.4567
A2*M1*M2         1    0.5   329    0.99542212    1.0118     3        660  0.3869
DSAMP*M1*M2      2    0     329    0.98707051    1.4362     6       1320  0.1972
KB_B*M1*M2       2    0     329    0.99253218    0.8261     6       1320  0.5496
MD               1    0.5   329    0.99385039    1.3613     3        660  0.2535
A1*MD            1    0.5   329    0.99508713    1.0862     3        660  0.3542
A2*MD            1    0.5   329    0.99691506    0.6808     3        660  0.5640
A1*A2*MD         1    0.5   329    0.99861500    0.3051     3        660  0.8217
DSAMP*MD         2    0     329    0.99202206    0.8829     6       1320  0.5066
A1*DSAMP*MD      2    0     329    0.98588731    1.5690     6       1320  0.1526
A2*DSAMP*MD      2    0     329    0.99097500    0.9995     6       1320  0.4240
KB_B*MD          2    0     329    0.99248868    0.8309     6       1320  0.5459
A1*KB_B*MD       2    0     329    0.99252820    0.8265     6       1320  0.5493
A2*KB_B*MD       2    0     329    0.99733617    0.2936     6       1320  0.9401
DSAMP*KB_B*MD    3    0     329    0.97973229    1.1307    12   1746.487  0.3300
M1*MD            1    0.5   329    0.99669015    0.7306     3        660  0.5340
A1*M1*MD         1    0.5   329    0.99717107    0.6241     3        660  0.5996
A2*M1*MD         1    0.5   329    0.99556697    0.9796     3        660  0.4018
DSAMP*M1*MD      2    0     329    0.99072793    1.0271     6       1320  0.4058
KB_B*M1*MD       2    0     329    0.99042310    1.0611     6       1320  0.3840
M2*MD            1    0.5   329    0.99620953    0.8371     3        660  0.4738
A1*M2*MD         1    0.5   329    0.99487525    1.1333     3        660  0.3348
A2*M2*MD         1    0.5   329    0.99431914    1.2569     3        660  0.2882
DSAMP*M2*MD      2    0     329    0.99377068    0.6884     6       1320  0.6590
KB_B*M2*MD       2    0     329    0.98936364    1.1794     6       1320  0.3146
M1*M2*MD         1    0.5   329    0.99700092    0.6618     3        660  0.5758
AGE              2    0     329    0.95938521    4.6085     6       1320  0.0001
A1*AGE           2    0     329    0.98024804    2.2054     6       1320  0.0402
A2*AGE           2    0     329    0.98702699    1.4411     6       1320  0.1954
A1*A2*AGE        2    0     329    0.99119379    0.9751     6       1320  0.4406
DSAMP*AGE        3    0     329    0.98745328    0.6962    12   1746.487  0.7565
A1*DSAMP*AGE     3    0     329    0.98536943    0.8130    12   1746.487  0.6373
```

```
A2*DSAMP*AGE     3    0    329   0.98584169     0.7865    12  1746.487  0.6650
KB_B*AGE         3    0    329   0.95621229     2.4840    12  1746.487  0.0032
A1*KB_B*AGE      3    0    329   0.94482782     3.1556    12  1746.487  0.0002
A2*KB_B*AGE      3    0    329   0.98039300     1.0934    12  1746.487  0.3612
DSAMP*KB_B*AGE   3    2    329   0.97118097     0.8085    24  1914.802  0.7294
M1*AGE           2    0    329   0.98836806     1.2908     6      1320  0.2583
A1*M1*AGE        2    0    329   0.98934749     1.1812     6      1320  0.3136
A2*M1*AGE        2    0    329   0.99720740     0.3078     6      1320  0.9331
DSAMP*M1*AGE     3    0    329   0.98805321     0.6626    12  1746.487  0.7885
KB_B*M1*AGE      3    0    329   0.97115067     1.6193    12  1746.487  0.0798
M2*AGE           2    0    329   0.97815681     2.4428     6      1320  0.0236
A1*M2*AGE        2    0    329   0.99100771     0.9959     6      1320  0.4265
A2*M2*AGE        2    0    329   0.98892376     1.2286     6      1320  0.2886
DSAMP*M2*AGE     3    0    329   0.98243461     0.9781    12  1746.487  0.4675
KB_B*M2*AGE      3    0    329   0.97704133     1.2833    12  1746.487  0.2215
M1*M2*AGE        2    0    329   0.97907009     2.3391     6      1320  0.0298
MD*AGE           2    0    329   0.99392601     0.6712     6      1320  0.6730
A1*MD*AGE        2    0    329   0.99120433     0.9740     6      1320  0.4414
A2*MD*AGE        2    0    329   0.98723872     1.4173     6      1320  0.2044
DSAMP*MD*AGE     3    0    329   0.97719156     1.2748    12  1746.487  0.2268
KB_B*MD*AGE      3    0    329   0.99238246     0.4212    12  1746.487  0.9559
M1*MD*AGE        2    0    329   0.99574147     0.4699     6      1320  0.8310
M2*MD*AGE        2    0    329   0.99473747     0.5812     6      1320  0.7456
------------------------------------------------------------------------------

Dependent Variable: LOGERR_0
                                         Sum of           Mean
Source                     DF           Squares         Square   F Value     Pr > F
Model                     201       22.80255867     0.11344557      2.99     0.0001
Error                     662       25.12022448     0.03794596
Corrected Total           863       47.92278315

                     R-Square              C.V.       Root MSE          LOGERR_0 Mean
                     0.475819          4.852491       0.194797               4.014376

Source                     DF          Anova SS    Mean Square   F Value     Pr > F

A1                          1        1.63188633     1.63188633     43.01     0.0001
A2                          1        2.40289422     2.40289422     63.32     0.0001
A1*A2                       1        0.12411475     0.12411475      3.27     0.0710
DSAMP                       2        0.09472232     0.04736116      1.25     0.2877
A1*DSAMP                    2        0.03071706     0.01535853      0.40     0.6673
A2*DSAMP                    2        0.01200406     0.00600203      0.16     0.8537
A1*A2*DSAMP                 2        0.03001006     0.01500503      0.40     0.6735
KB_B                        2        3.42193520     1.71096760     45.09     0.0001
A1*KB_B                     2        1.34555813     0.67277907     17.73     0.0001
A2*KB_B                     2        1.32267763     0.66133882     17.43     0.0001
A1*A2*KB_B                  2        1.16337761     0.58168880     15.33     0.0001
DSAMP*KB_B                  4        0.15777758     0.03944439      1.04     0.3859
A1*DSAMP*KB_B               4        0.19103745     0.04775936      1.26     0.2849
```

```
A2*DSAMP*KB_B            4      0.05272534     0.01318134       0.35      0.8459
M1                       1      0.11530926     0.11530926       3.04      0.0818
A1*M1                    1      0.11781549     0.11781549       3.10      0.0785
A2*M1                    1      0.04486989     0.04486989       1.18      0.2772
A1*A2*M1                 1      0.03355667     0.03355667       0.88      0.3474
DSAMP*M1                 2      0.04767416     0.02383708       0.63      0.5339
A1*DSAMP*M1              2      0.03538315     0.01769158       0.47      0.6276
A2*DSAMP*M1              2      0.03506885     0.01753442       0.46      0.6302
KB_B*M1                  2      0.30025162     0.15012581       3.96      0.0196
A1*KB_B*M1               2      0.36307095     0.18153548       4.78      0.0087
A2*KB_B*M1               2      0.14739380     0.07369690       1.94      0.1442
DSAMP*KB_B*M1            4      0.06459973     0.01614993       0.43      0.7902
M2                       1      0.39532250     0.39532250      10.42      0.0013
A1*M2                    1      0.01170401     0.01170401       0.31      0.5788
A2*M2                    1      0.08335156     0.08335156       2.20      0.1388
A1*A2*M2                 1      0.01804012     0.01804012       0.48      0.4907
DSAMP*M2                 2      0.20527806     0.10263903       2.70      0.0676
A1*DSAMP*M2              2      0.12222706     0.06111353       1.61      0.2006
A2*DSAMP*M2              2      0.25402772     0.12701386       3.35      0.0358
KB_B*M2                  2      0.38766585     0.19383293       5.11      0.0063
A1*KB_B*M2               2      0.01648354     0.00824177       0.22      0.8048
A2*KB_B*M2               2      0.21393642     0.10696821       2.82      0.0604
DSAMP*KB_B*M2            4      0.20448627     0.05112157       1.35      0.2509
M1*M2                    1      0.12547692     0.12547692       3.31      0.0694
A1*M1*M2                 1      0.09109895     0.09109895       2.40      0.1218
A2*M1*M2                 1      0.00205056     0.00205056       0.05      0.8163
DSAMP*M1*M2              2      0.05110972     0.02555486       0.67      0.5103
KB_B*M1*M2               2      0.15004112     0.07502056       1.98      0.1393
MD                       1      0.03900841     0.03900841       1.03      0.3110
A1*MD                    1      0.01965024     0.01965024       0.52      0.4720
A2*MD                    1      0.00761344     0.00761344       0.20      0.6544
A1*A2*MD                 1      0.00892467     0.00892467       0.24      0.6279
DSAMP*MD                 2      0.02229935     0.01114968       0.29      0.7455
A1*DSAMP*MD              2      0.13595922     0.06797961       1.79      0.1675
A2*DSAMP*MD              2      0.08413205     0.04206603       1.11      0.3306
KB_B*MD                  2      0.06960737     0.03480368       0.92      0.4001
A1*KB_B*MD               2      0.05928690     0.02964345       0.78      0.4583
A2*KB_B*MD               2      0.01162396     0.00581198       0.15      0.8580
DSAMP*KB_B*MD            4      0.10522053     0.02630513       0.69      0.5968
M1*MD                    1      0.01038433     0.01038433       0.27      0.6011
A1*M1*MD                 1      0.00211622     0.00211622       0.06      0.8134
A2*M1*MD                 1      0.01139543     0.01139543       0.30      0.5839
DSAMP*M1*MD              2      0.00148493     0.00074246       0.02      0.9806
KB_B*M1*MD               2      0.05187010     0.02593505       0.68      0.5052
M2*MD                    1      0.02508252     0.02508252       0.66      0.4165
A1*M2*MD                 1      0.02243381     0.02243381       0.59      0.4422
A2*M2*MD                 1      0.11548860     0.11548860       3.04      0.0815
DSAMP*M2*MD              2      0.14164536     0.07082268       1.87      0.1555
KB_B*M2*MD               2      0.06736949     0.03368475       0.89      0.4121
M1*M2*MD                 1      0.05893561     0.05893561       1.55      0.2131
```

```
AGE                      2      0.60919685     0.30459842      8.03     0.0004
A1*AGE                   2      0.36863886     0.18431943      4.86     0.0080
A2*AGE                   2      0.03453056     0.01726528      0.45     0.6346
A1*A2*AGE                2      0.00807522     0.00403761      0.11     0.8991
DSAMP*AGE                4      0.15933750     0.03983437      1.05     0.3806
A1*DSAMP*AGE             4      0.18439376     0.04609844      1.21     0.3032
A2*DSAMP*AGE             4      0.02592923     0.00648231      0.17     0.9533
KB_B*AGE                 4      0.73886102     0.18471526      4.87     0.0007
A1*KB_B*AGE              4      1.01586496     0.25396624      6.69     0.0001
A2*KB_B*AGE              4      0.06297287     0.01574322      0.41     0.7980
DSAMP*KB_B*AGE           8      0.37001141     0.04625143      1.22     0.2850
M1*AGE                   2      0.23185100     0.11592550      3.06     0.0478
A1*M1*AGE                2      0.20270648     0.10135324      2.67     0.0699
A2*M1*AGE                2      0.04245268     0.02122634      0.56     0.5718
DSAMP*M1*AGE             4      0.05672642     0.01418160      0.37     0.8274
KB_B*M1*AGE              4      0.68831658     0.17207914      4.53     0.0013
M2*AGE                   2      0.17678739     0.08839369      2.33     0.0981
A1*M2*AGE                2      0.06858832     0.03429416      0.90     0.4055
A2*M2*AGE                2      0.22316212     0.11158106      2.94     0.0535
DSAMP*M2*AGE             4      0.08822261     0.02205565      0.58     0.6763
KB_B*M2*AGE              4      0.18307407     0.04576852      1.21     0.3069
M1*M2*AGE                2      0.02150361     0.01075180      0.28     0.7534
MD*AGE                   2      0.04465692     0.02232846      0.59     0.5555
A1*MD*AGE                2      0.11698131     0.05849066      1.54     0.2148
A2*MD*AGE                2      0.13398830     0.06699415      1.77     0.1719
DSAMP*MD*AGE             4      0.10431711     0.02607928      0.69     0.6009
KB_B*MD*AGE              4      0.07858536     0.01964634      0.52     0.7227
M1*MD*AGE                2      0.01714596     0.00857298      0.23     0.7978
M2*MD*AGE                2      0.05541594     0.02770797      0.73     0.4822
------------------------------------------------------------------------------

Dependent Variable: LOGERR_1
                                    Sum of           Mean
Source                  DF         Squares         Square  F Value     Pr > F
Model                  201     33.15737263     0.16496205      4.11     0.0001
Error                  662     26.58954596     0.04016548
Corrected Total        863     59.74691859

                  R-Square            C.V.       Root MSE        LOGERR_1 Mean
                  0.554964        4.955245       0.200413             4.044467

Source                  DF        Anova SS    Mean Square  F Value     Pr > F
A1                       1      3.33131459     3.33131459     82.94     0.0001
A2                       1      5.78589464     5.78589464    144.05     0.0001
A1*A2                    1      0.84568474     0.84568474     21.06     0.0001
DSAMP                    2      0.08021050     0.04010525      1.00     0.3690
A1*DSAMP                 2      0.07210133     0.03605067      0.90     0.4081
A2*DSAMP                 2      0.24613289     0.12306644      3.06     0.0474
A1*A2*DSAMP              2      0.19698842     0.09849421      2.45     0.0869
KB_B                     2      5.73858764     2.86929382     71.44     0.0001
```

```
A1*KB_B                 2       3.38808521     1.69404260      42.18       0.0001
A2*KB_B                 2       2.90320612     1.45160306      36.14       0.0001
A1*A2*KB_B              2       3.37578424     1.68789212      42.02       0.0001
DSAMP*KB_B              4       0.04762751     0.01190688       0.30       0.8803
A1*DSAMP*KB_B           4       0.05573667     0.01393417       0.35       0.8462
A2*DSAMP*KB_B           4       0.39089602     0.09772401       2.43       0.0463
M1                      1       0.10305426     0.10305426       2.57       0.1097
A1*M1                   1       0.13398836     0.13398836       3.34       0.0682
A2*M1                   1       0.06021386     0.06021386       1.50       0.2212
A1*A2*M1                1       0.08433720     0.08433720       2.10       0.1478
DSAMP*M1                2       0.03633511     0.01816756       0.45       0.6363
A1*DSAMP*M1             2       0.01330552     0.00665276       0.17       0.8474
A2*DSAMP*M1             2       0.00897302     0.00448651       0.11       0.8943
KB_B*M1                 2       0.32853834     0.16426917       4.09       0.0172
A1*KB_B*M1              2       0.28900117     0.14450058       3.60       0.0279
A2*KB_B*M1              2       0.10720687     0.05360343       1.33       0.2640
DSAMP*KB_B*M1           4       0.02170692     0.00542673       0.14       0.9694
M2                      1       0.03390249     0.03390249       0.84       0.3586
A1*M2                   1       0.00240645     0.00240645       0.06       0.8067
A2*M2                   1       0.02385312     0.02385312       0.59       0.4412
A1*A2*M2                1       0.08381906     0.08381906       2.09       0.1490
DSAMP*M2                2       0.27095954     0.13547977       3.37       0.0349
A1*DSAMP*M2             2       0.31113282     0.15556641       3.87       0.0213
A2*DSAMP*M2             2       0.07411122     0.03705561       0.92       0.3980
KB_B*M2                 2       0.08282466     0.04141233       1.03       0.3572
A1*KB_B*M2              2       0.00952633     0.00476316       0.12       0.8882
A2*KB_B*M2              2       0.01335869     0.00667935       0.17       0.8468
DSAMP*KB_B*M2           4       0.41785098     0.10446274       2.60       0.0351
M1*M2                   1       0.09615214     0.09615214       2.39       0.1223
A1*M1*M2                1       0.07025708     0.07025708       1.75       0.1864
A2*M1*M2                1       0.00035061     0.00035061       0.01       0.9256
DSAMP*M1*M2             2       0.25476412     0.12738206       3.17       0.0426
KB_B*M1*M2              2       0.10889934     0.05444967       1.36       0.2585
MD                      1       0.00689900     0.00689900       0.17       0.6787
A1*MD                   1       0.01640546     0.01640546       0.41       0.5230
A2*MD                   1       0.05413279     0.05413279       1.35       0.2461
A1*A2*MD                1       0.03520914     0.03520914       0.88       0.3495
DSAMP*MD                2       0.01343831     0.00671915       0.17       0.8460
A1*DSAMP*MD             2       0.01660913     0.00830457       0.21       0.8133
A2*DSAMP*MD             2       0.05477006     0.02738503       0.68       0.5061
KB_B*MD                 2       0.00585254     0.00292627       0.07       0.9297
A1*KB_B*MD              2       0.00824020     0.00412010       0.10       0.9025
A2*KB_B*MD              2       0.04194141     0.02097070       0.52       0.5935
DSAMP*KB_B*MD           4       0.03944795     0.00986199       0.25       0.9124
M1*MD                   1       0.01593493     0.01593493       0.40       0.5290
A1*M1*MD                1       0.02932903     0.02932903       0.73       0.3931
A2*M1*MD                1       0.08321608     0.08321608       2.07       0.1505
DSAMP*M1*MD             2       0.02020144     0.01010072       0.25       0.7777
KB_B*M1*MD              2       0.03407556     0.01703778       0.42       0.6545
M2*MD                   1       0.00911608     0.00911608       0.23       0.6339
```

```
A1*M2*MD                 1      0.01974073     0.01974073      0.49      0.4835
A2*M2*MD                 1      0.02153506     0.02153506      0.54      0.4643
DSAMP*M2*MD              2      0.08807697     0.04403849      1.10      0.3347
KB_B*M2*MD               2      0.00868348     0.00434174      0.11      0.8976
M1*M2*MD                 1      0.01009416     0.01009416      0.25      0.6163
AGE                      2      0.08977290     0.04488645      1.12      0.3277
A1*AGE                   2      0.08977290     0.04488645      1.12      0.3277
A2*AGE                   2      0.00599799     0.00299899      0.07      0.9281
A1*A2*AGE                2      0.00599799     0.00299899      0.07      0.9281
DSAMP*AGE                4      0.08226073     0.02056518      0.51      0.7269
A1*DSAMP*AGE             4      0.07049397     0.01762349      0.44      0.7806
A2*DSAMP*AGE             4      0.07015940     0.01753985      0.44      0.7821
KB_B*AGE                 4      0.13147620     0.03286905      0.82      0.5137
A1*KB_B*AGE              4      0.13147621     0.03286905      0.82      0.5137
A2*KB_B*AGE              4      0.00559201     0.00139800      0.03      0.9977
DSAMP*KB_B*AGE           8      0.22382085     0.02797761      0.70      0.6948
M1*AGE                   2      0.16986613     0.08493306      2.11      0.1215
A1*M1*AGE                2      0.16568048     0.08284024      2.06      0.1280
A2*M1*AGE                2      0.00063316     0.00031658      0.01      0.9921
DSAMP*M1*AGE             4      0.11808031     0.02952008      0.73      0.5683
KB_B*M1*AGE              4      0.25050739     0.06262685      1.56      0.1835
M2*AGE                   2      0.07084310     0.03542155      0.88      0.4145
A1*M2*AGE                2      0.07084310     0.03542155      0.88      0.4145
A2*M2*AGE                2      0.16180050     0.08090025      2.01      0.1342
DSAMP*M2*AGE             4      0.21387895     0.05346974      1.33      0.2568
KB_B*M2*AGE              4      0.11429327     0.02857332      0.71      0.5843
M1*M2*AGE                2      0.12765615     0.06382808      1.59      0.2049
MD*AGE                   2      0.04513245     0.02256622      0.56      0.5704
A1*MD*AGE                2      0.02116108     0.01058054      0.26      0.7685
A2*MD*AGE                2      0.17468882     0.08734441      2.17      0.1145
DSAMP*MD*AGE             4      0.24050280     0.06012570      1.50      0.2014
KB_B*MD*AGE              4      0.06619601     0.01654900      0.41      0.8000
M1*MD*AGE                2      0.00161182     0.00080591      0.02      0.9801
M2*MD*AGE                2      0.10514665     0.05257332      1.31      0.2708
------------------------------------------------------------------------------

Dependent Variable: LOGERR_2
                                    Sum of           Mean
Source                  DF         Squares         Square   F Value      Pr > F
Model                  201     20.30601479     0.10102495      4.28      0.0001
Error                  662     15.61477296     0.02358727
Corrected Total        863     35.92078775

                  R-Square            C.V.       Root MSE          LOGERR_2 Mean
                  0.565300        3.824324       0.153581               4.015911

Source                  DF        Anova SS    Mean Square   F Value      Pr > F
A1                       1      1.54700811     1.54700811     65.59      0.0001
A2                       1      2.45564240     2.45564240    104.11      0.0001
A1*A2                    1      0.11507709     0.11507709      4.88      0.0275
```

```
DSAMP               2      0.10476965     0.05238482      2.22     0.1093
A1*DSAMP            2      0.04508240     0.02254120      0.96     0.3851
A2*DSAMP            2      0.10140645     0.05070322      2.15     0.1173
A1*A2*DSAMP         2      0.13588029     0.06794014      2.88     0.0568
KB_B                2      3.63065435     1.81532718     76.96     0.0001
A1*KB_B             2      1.28732466     0.64366233     27.29     0.0001
A2*KB_B             2      1.36254180     0.68127090     28.88     0.0001
A1*A2*KB_B          2      1.24086133     0.62043066     26.30     0.0001
DSAMP*KB_B          4      0.07500093     0.01875023      0.79     0.5287
A1*DSAMP*KB_B       4      0.07429743     0.01857436      0.79     0.5335
A2*DSAMP*KB_B       4      0.24480731     0.06120183      2.59     0.0355
M1                  1      0.14118510     0.14118510      5.99     0.0147
A1*M1               1      0.10146761     0.10146761      4.30     0.0385
A2*M1               1      0.05199971     0.05199971      2.20     0.1381
A1*A2*M1            1      0.02918227     0.02918227      1.24     0.2664
DSAMP*M1            2      0.01868331     0.00934166      0.40     0.6731
A1*DSAMP*M1         2      0.00683287     0.00341643      0.14     0.8652
A2*DSAMP*M1         2      0.00687462     0.00343731      0.15     0.8644
KB_B*M1             2      0.29380459     0.14690229      6.23     0.0021
A1*KB_B*M1          2      0.34255445     0.17127723      7.26     0.0008
A2*KB_B*M1          2      0.11115081     0.05557541      2.36     0.0956
DSAMP*KB_B*M1       4      0.02786803     0.00696701      0.30     0.8810
M2                  1      0.18727868     0.18727868      7.94     0.0050
A1*M2               1      0.00023600     0.00023600      0.01     0.9204
A2*M2               1      0.16802705     0.16802705      7.12     0.0078
A1*A2*M2            1      0.00005601     0.00005601      0.00     0.9612
DSAMP*M2            2      0.37997533     0.18998767      8.05     0.0003
A1*DSAMP*M2         2      0.25492606     0.12746303      5.40     0.0047
A2*DSAMP*M2         2      0.11068369     0.05534185      2.35     0.0965
KB_B*M2             2      0.43969349     0.21984675      9.32     0.0001
A1*KB_B*M2          2      0.01027301     0.00513650      0.22     0.8044
A2*KB_B*M2          2      0.11871821     0.05935910      2.52     0.0815
DSAMP*KB_B*M2       4      0.35758053     0.08939513      3.79     0.0047
M1*M2               1      0.02795582     0.02795582      1.19     0.2767
A1*M1*M2            1      0.04001633     0.04001633      1.70     0.1932
A2*M1*M2            1      0.02795582     0.02795582      1.19     0.2767
DSAMP*M1*M2         2      0.06809075     0.03404537      1.44     0.2369
KB_B*M1*M2          2      0.04017977     0.02008989      0.85     0.4271
MD                  1      0.02633695     0.02633695      1.12     0.2910
A1*MD               1      0.01104189     0.01104189      0.47     0.4941
A2*MD               1      0.02633695     0.02633695      1.12     0.2910
A1*A2*MD            1      0.01104189     0.01104189      0.47     0.4941
DSAMP*MD            2      0.03644863     0.01822431      0.77     0.4622
A1*DSAMP*MD         2      0.09961949     0.04980974      2.11     0.1218
A2*DSAMP*MD         2      0.04222237     0.02111119      0.90     0.4091
KB_B*MD             2      0.02754227     0.01377113      0.58     0.5580
A1*KB_B*MD          2      0.04227342     0.02113671      0.90     0.4087
A2*KB_B*MD          2      0.02754227     0.01377113      0.58     0.5580
DSAMP*KB_B*MD       4      0.12188044     0.03047011      1.29     0.2718
M1*MD               1      0.00454238     0.00454238      0.19     0.6609
```

```
A1*M1*MD                 1      0.00010386     0.00010386      0.00      0.9471
A2*M1*MD                 1      0.03638305     0.03638305      1.54      0.2147
DSAMP*M1*MD              2      0.02521243     0.01260621      0.53      0.5862
KB_B*M1*MD               2      0.01446903     0.00723451      0.31      0.7360
M2*MD                    1      0.00008827     0.00008827      0.00      0.9512
A1*M2*MD                 1      0.00054971     0.00054971      0.02      0.8787
A2*M2*MD                 1      0.01913146     0.01913146      0.81      0.3681
DSAMP*M2*MD              2      0.07323079     0.03661539      1.55      0.2125
KB_B*M2*MD               2      0.00208755     0.00104378      0.04      0.9567
M1*M2*MD                 1      0.00893015     0.00893015      0.38      0.5386
AGE                      2      0.46996145     0.23498072      9.96      0.0001
A1*AGE                   2      0.23400976     0.11700488      4.96      0.0073
A2*AGE                   2      0.07591350     0.03795675      1.61      0.2008
A1*A2*AGE                2      0.01750630     0.00875315      0.37      0.6901
DSAMP*AGE                4      0.08170012     0.02042503      0.87      0.4840
A1*DSAMP*AGE             4      0.08746145     0.02186536      0.93      0.4477
A2*DSAMP*AGE             4      0.04718887     0.01179722      0.50      0.7356
KB_B*AGE                 4      0.38118307     0.09529577      4.04      0.0030
A1*KB_B*AGE              4      0.47437883     0.11859471      5.03      0.0005
A2*KB_B*AGE              4      0.06623141     0.01655785      0.70      0.5908
DSAMP*KB_B*AGE           8      0.23961675     0.02995209      1.27      0.2562
M1*AGE                   2      0.10930050     0.05465025      2.32      0.0994
A1*M1*AGE                2      0.08914860     0.04457430      1.89      0.1519
A2*M1*AGE                2      0.00971519     0.00485759      0.21      0.8139
DSAMP*M1*AGE             4      0.06454034     0.01613509      0.68      0.6032
KB_B*M1*AGE              4      0.28414945     0.07103736      3.01      0.0177
M2*AGE                   2      0.26823859     0.13411930      5.69      0.0036
A1*M2*AGE                2      0.11571870     0.05785935      2.45      0.0868
A2*M2*AGE                2      0.15913453     0.07956726      3.37      0.0349
DSAMP*M2*AGE             4      0.11642313     0.02910578      1.23      0.2951
KB_B*M2*AGE              4      0.23091686     0.05772921      2.45      0.0452
M1*M2*AGE                2      0.04812077     0.02406039      1.02      0.3611
MD*AGE                   2      0.01133936     0.00566968      0.24      0.7864
A1*MD*AGE                2      0.01136253     0.00568126      0.24      0.7860
A2*MD*AGE                2      0.01861537     0.00930768      0.39      0.6741
DSAMP*MD*AGE             4      0.10466782     0.02616695      1.11      0.3510
KB_B*MD*AGE              4      0.01327888     0.00331972      0.14      0.9670
M1*MD*AGE                2      0.00813762     0.00406881      0.17      0.8416
M2*MD*AGE                2      0.02743572     0.01371786      0.58      0.5593
-----------------------------------------------------------------------------
-----------------------------------------------------------------------------


KB_D=10

MULTIVARIATE ANALYSIS OF VARIANCE

Source         S    M     N    Wilk's Lambda     F       df1       df2  Pr > F

A1             1   0.5   398    0.96717879     9.0267      3       798  0.0001
```

```
A2             1   0.5   398    0.98104138     5.1404       3        798  0.0016
A1*A2          1   0.5   398    0.98231445     4.7891       3        798  0.0026
DSAMP          2    0    398    0.92261414    10.9310       6       1596  0.0001
A1*DSAMP       2    0    398    0.95529879     6.1523       6       1596  0.0001
A2*DSAMP       2    0    398    0.98877583     1.5055       6       1596  0.1725
KB_B           2    0    398    0.84223634    23.8442       6       1596  0.0001
A1*KB_B        2    0    398    0.94982372     6.9356       6       1596  0.0001
A2*KB_B        2    0    398    0.97189333     3.8189       6       1596  0.0009
DSAMP*KB_B     3    0    398    0.96476551     2.4019      12   2111.601  0.0044
M1             1   0.5   398    0.99884406     0.3078       3        798  0.8197
A1*M1          1   0.5   398    0.99957882     0.1121       3        798  0.9530
A2*M1          1   0.5   398    0.99695566     0.8123       3        798  0.4872
DSAMP*M1       2    0    398    0.99718065     0.3758       6       1596  0.8947
KB_B*M1        2    0    398    0.99441009     0.7466       6       1596  0.6122
M2             1   0.5   398    0.99634754     0.9751       3        798  0.4038
A1*M2          1   0.5   398    0.99462939     1.4363       3        798  0.2308
A2*M2          1   0.5   398    0.99619937     1.0148       3        798  0.3854
DSAMP*M2       2    0    398    0.98883998     1.4968       6       1596  0.1754
KB_B*M2        2    0    398    0.98618044     1.8573       6       1596  0.0848
M1*M2          1   0.5   398    0.99440952     1.4954       3        798  0.2144
MD             1   0.5   398    0.99978830     0.0563       3        798  0.9824
A1*MD          1   0.5   398    0.99751699     0.6621       3        798  0.5755
A2*MD          1   0.5   398    0.99591395     1.0913       3        798  0.3519
DSAMP*MD       2    0    398    0.99513528     0.6494       6       1596  0.6907
KB_B*MD        2    0    398    0.99193376     1.0793       6       1596  0.3725
M1*MD          1   0.5   398    0.99704763     0.7877       3        798  0.5009
M2*MD          1   0.5   398    0.99793382     0.5507       3        798  0.6478
AGE            2    0    398    0.98488816     2.0329       6       1596  0.0584
A1*AGE         2    0    398    0.98623677     1.8496       6       1596  0.0861
A2*AGE         2    0    398    0.99315817     0.9147       6       1596  0.4832
DSAMP*AGE      3    0    398    0.96621691     2.3006      12   2111.601  0.0066
KB_B*AGE       3    0    398    0.96732458     2.2234      12   2111.601  0.0089
M1*AGE         2    0    398    0.99383955     0.8231       6       1596  0.5519
M2*AGE         2    0    398    0.99377376     0.8320       6       1596  0.5451
MD*AGE         2    0    398    0.99165403     1.1170       6       1596  0.3498
-----------------------------------------------------------------------------

Dependent Variable: LOGERR_0
                                      Sum of           Mean
Source                  DF           Squares         Square   F Value     Pr > F
Model                   63       236.2300036      3.7496826      2.81     0.0001
Error                  800      1066.3567662      1.3329460
Corrected Total        863      1302.5867697

                  R-Square              C.V.       Root MSE        LOGERR_0 Mean
                  0.181355          50.13376       1.154533             2.302905

Source                  DF          Anova SS    Mean Square   F Value     Pr > F
A1                       1        5.81741430     5.81741430      4.36     0.0370
A2                       1        0.02249335     0.02249335      0.02     0.8967
```

```
A1*A2                    1      6.94454819     6.94454819      5.21     0.0227
DSAMP                    2     64.80792000    32.40396000     24.31     0.0001
A1*DSAMP                 2     14.11519862     7.05759931      5.29     0.0052
A2*DSAMP                 2      3.65935862     1.82967931      1.37     0.2540
KB_B                     2     33.28856053    16.64428027     12.49     0.0001
A1*KB_B                  2     12.69639599     6.34819800      4.76     0.0088
A2*KB_B                  2      9.85631975     4.92815987      3.70     0.0252
DSAMP*KB_B               4     21.76290229     5.44072557      4.08     0.0028
M1                       1      0.83994273     0.83994273      0.63     0.4275
A1*M1                    1      0.03380624     0.03380624      0.03     0.8735
A2*M1                    1      0.23611288     0.23611288      0.18     0.6740
DSAMP*M1                 2      2.21223282     1.10611641      0.83     0.4365
KB_B*M1                  2      4.10611801     2.05305901      1.54     0.2150
M2                       1      0.69619006     0.69619006      0.52     0.4701
A1*M2                    1      0.04415844     0.04415844      0.03     0.8556
A2*M2                    1      0.88250876     0.88250876      0.66     0.4161
DSAMP*M2                 2      0.82159983     0.41079992      0.31     0.7349
KB_B*M2                  2      3.03651031     1.51825516      1.14     0.3207
M1*M2                    1      0.11957698     0.11957698      0.09     0.7646
MD                       1      0.03631491     0.03631491      0.03     0.8689
A1*MD                    1      1.01565900     1.01565900      0.76     0.3830
A2*MD                    1      0.18011865     0.18011865      0.14     0.7133
DSAMP*MD                 2      1.31173329     0.65586664      0.49     0.6116
KB_B*MD                  2      0.00440011     0.00220006      0.00     0.9984
M1*MD                    1      1.02263761     1.02263761      0.77     0.3813
M2*MD                    1      0.02851821     0.02851821      0.02     0.8837
AGE                      2     13.24884419     6.62442210      4.97     0.0072
A1*AGE                   2      3.47205639     1.73602820      1.30     0.2725
A2*AGE                   2      0.15566801     0.07783401      0.06     0.9433
DSAMP*AGE                4     19.64330145     4.91082536      3.68     0.0056
KB_B*AGE                 4      1.53616423     0.38404106      0.29     0.8858
M1*AGE                   2      1.76779282     0.88389641      0.66     0.5155
M2*AGE                   2      5.39503140     2.69751570      2.02     0.1328
MD*AGE                   2      1.41189454     0.70594727      0.53     0.5890
------------------------------------------------------------------------------

Dependent Variable: LOGERR_1
                                    Sum of           Mean
Source                  DF         Squares         Square   F Value     Pr > F
Model                   63     248.6509701      3.9468408      4.88     0.0001
Error                  800     647.2158039      0.8090198
Corrected Total        863     895.8667740

                  R-Square            C.V.       Root MSE        LOGERR_1 Mean
                  0.277554        39.96715       0.899455             2.250486

Source                  DF        Anova SS    Mean Square   F Value     Pr > F
A1                       1     20.14878261    20.14878261     24.91     0.0001
A2                       1      4.94065466     4.94065466      6.11     0.0137
A1*A2                    1      7.66483963     7.66483963      9.47     0.0022
```

```
DSAMP                    2     32.30515092    16.15257546     19.97     0.0001
A1*DSAMP                 2     27.76022013    13.88011007     17.16     0.0001
A2*DSAMP                 2      5.43390362     2.71695181      3.36     0.0353
KB_B                     2     48.14435272    24.07217636     29.75     0.0001
A1*KB_B                  2     26.10142515    13.05071257     16.13     0.0001
A2*KB_B                  2     15.45778916     7.72889458      9.55     0.0001
DSAMP*KB_B               4     11.98267839     2.99566960      3.70     0.0054
M1                       1      0.53175984     0.53175984      0.66     0.4178
A1*M1                    1      0.03671717     0.03671717      0.05     0.8314
A2*M1                    1      1.40500832     1.40500832      1.74     0.1879
DSAMP*M1                 2      0.37400676     0.18700338      0.23     0.7937
KB_B*M1                  2      0.85802891     0.42901445      0.53     0.5886
M2                       1      2.19325981     2.19325981      2.71     0.1001
A1*M2                    1      0.37513004     0.37513004      0.46     0.4961
A2*M2                    1      0.01838648     0.01838648      0.02     0.8802
DSAMP*M2                 2      0.10116185     0.05058093      0.06     0.9394
KB_B*M2                  2      2.40707438     1.20353719      1.49     0.2265
M1*M2                    1      2.60170500     2.60170500      3.22     0.0733
MD                       1      0.02645353     0.02645353      0.03     0.8565
A1*MD                    1      0.77850618     0.77850618      0.96     0.3269
A2*MD                    1      0.99882386     0.99882386      1.23     0.2668
DSAMP*MD                 2      1.31648409     0.65824204      0.81     0.4436
KB_B*MD                  2      2.73549021     1.36774511      1.69     0.1851
M1*MD                    1      0.12881665     0.12881665      0.16     0.6900
M2*MD                    1      0.66824471     0.66824471      0.83     0.3637
AGE                      2      6.38460600     3.19230300      3.95     0.0197
A1*AGE                   2      2.09533205     1.04766602      1.29     0.2745
A2*AGE                   2      0.18375729     0.09187864      0.11     0.8927
DSAMP*AGE                4     12.21332167     3.05333042      3.77     0.0048
KB_B*AGE                 4      6.94630457     1.73657614      2.15     0.0734
M1*AGE                   2      0.45277503     0.22638752      0.28     0.7560
M2*AGE                   2      2.19324664     1.09662332      1.36     0.2584
MD*AGE                   2      0.68677206     0.34338603      0.42     0.6543
------------------------------------------------------------------------------

Dependent Variable: LOGERR_2
                                    Sum of           Mean
Source                  DF         Squares         Square   F Value     Pr > F
Model                   63     203.3232589      3.2273533      3.94     0.0001
Error                  800     655.6469120      0.8195586
Corrected Total        863     858.9701709

                  R-Square            C.V.       Root MSE         LOGERR_2 Mean
                  0.236706        43.12634       0.905295              2.099169

Source                  DF        Anova SS    Mean Square   F Value     Pr > F
A1                       1      4.36257342     4.36257342      5.32     0.0213
A2                       1      3.92729477     3.92729477      4.79     0.0289
A1*A2                    1      9.57983528     9.57983528     11.69     0.0007
DSAMP                    2     40.21438114    20.10719057     24.53     0.0001
```

```
A1*DSAMP                 2      2.96204711     1.48102356       1.81      0.1648
A2*DSAMP                 2      2.38829384     1.19414692       1.46      0.2335
KB_B                     2     55.50833763    27.75416882      33.86      0.0001
A1*KB_B                  2     12.73107226     6.36553613       7.77      0.0005
A2*KB_B                  2      9.23810352     4.61905176       5.64      0.0037
DSAMP*KB_B               4     10.66965869     2.66741467       3.25      0.0116
M1                       1      0.62741957     0.62741957       0.77      0.3819
A1*M1                    1      0.04930975     0.04930975       0.06      0.8063
A2*M1                    1      0.69624202     0.69624202       0.85      0.3570
DSAMP*M1                 2      1.13836377     0.56918188       0.69      0.4996
KB_B*M1                  2      2.96457559     1.48228780       1.81      0.1645
M2                       1      0.56745094     0.56745094       0.69      0.4056
A1*M2                    1      1.03340674     1.03340674       1.26      0.2618
A2*M2                    1      0.22280249     0.22280249       0.27      0.6022
DSAMP*M2                 2      1.83145014     0.91572507       1.12      0.3277
KB_B*M2                  2      2.94024845     1.47012422       1.79      0.1670
M1*M2                    1      0.11976247     0.11976247       0.15      0.7024
MD                       1      0.00619049     0.00619049       0.01      0.9308
A1*MD                    1      0.00016057     0.00016057       0.00      0.9888
A2*MD                    1      0.01915391     0.01915391       0.02      0.8785
DSAMP*MD                 2      1.80030510     0.90015255       1.10      0.3339
KB_B*MD                  2      0.10172833     0.05086417       0.06      0.9398
M1*MD                    1      0.06732590     0.06732590       0.08      0.7745
M2*MD                    1      0.45325117     0.45325117       0.55      0.4573
AGE                      2      6.31976409     3.15988205       3.86      0.0216
A1*AGE                   2      5.03606922     2.51803461       3.07      0.0469
A2*AGE                   2      1.86085622     0.93042811       1.14      0.3218
DSAMP*AGE                4     13.29211567     3.32302892       4.05      0.0029
KB_B*AGE                 4      4.69416387     1.17354097       1.43      0.2215
M1*AGE                   2      2.92348814     1.46174407       1.78      0.1687
M2*AGE                   2      1.64582573     0.82291287       1.00      0.3668
MD*AGE                   2      1.33023095     0.66511547       0.81      0.4445
------------------------------------------------------------------------------
------------------------------------------------------------------------------


KB_D=100

MULTIVARIATE ANALYSIS OF VARIANCE

Source        S    M     N    Wilk's Lambda     F       df1       df2  Pr > F

A1            1   0.5   398    0.92857488    20.4605      3       798  0.0001
A2            1   0.5   398    0.99280980     1.9264      3       798  0.1238
A1*A2         1   0.5   398    0.99883958     0.3090      3       798  0.8189
DSAMP         2    0    398    0.98078530     2.5930      6      1596  0.0167
A1*DSAMP      2    0    398    0.99047301     1.2762      6      1596  0.2650
A2*DSAMP      2    0    398    0.99390242     0.8147      6      1596  0.5584
KB_B          2    0    398    0.60839814    75.0262      6      1596  0.0001
A1*KB_B       2    0    398    0.97097052     3.9471      6      1596  0.0006
```

```
A2*KB_B        2    0    398    0.98696721     1.7505      6       1596  0.1058
DSAMP*KB_B     3    0    398    0.98475882     1.0245     12   2111.601  0.4229
M1             1  0.5    398    0.99458925     1.4471      3        798  0.2278
A1*M1          1  0.5    398    0.99694841     0.8142      3        798  0.4862
A2*M1          1  0.5    398    0.99523122     1.2746      3        798  0.2819
DSAMP*M1       2    0    398    0.99415744     0.7805      6       1596  0.5852
KB_B*M1        2    0    398    0.99599542     0.5342      6       1596  0.7826
M2             1  0.5    398    0.99583985     1.1112      3        798  0.3437
A1*M2          1  0.5    398    0.99952587     0.1262      3        798  0.9446
A2*M2          1  0.5    398    0.99579541     1.1231      3        798  0.3388
DSAMP*M2       2    0    398    0.99816068     0.2450      6       1596  0.9614
KB_B*M2        2    0    398    0.98778680     1.6394      6       1596  0.1325
M1*M2          1  0.5    398    0.99014650     2.6471      3        798  0.0480
MD             1  0.5    398    0.99108369     2.3931      3        798  0.0672
A1*MD          1  0.5    398    0.99908350     0.2440      3        798  0.8656
A2*MD          1  0.5    398    0.99720728     0.7449      3        798  0.5255
DSAMP*MD       2    0    398    0.98632727     1.8373      6       1596  0.0884
KB_B*MD        2    0    398    0.99196455     1.0752      6       1596  0.3751
M1*MD          1  0.5    398    0.99291881     1.8970      3        798  0.1286
M2*MD          1  0.5    398    0.99685114     0.8402      3        798  0.4720
AGE            2    0    398    0.98994743     1.3472      6       1596  0.2328
A1*AGE         2    0    398    0.97683837     3.1351      6       1596  0.0047
A2*AGE         2    0    398    0.97871454     2.8770      6       1596  0.0086
DSAMP*AGE      3    0    398    0.98323615     1.1280     12   2111.601  0.3321
KB_B*AGE       3    0    398    0.93796844     4.3111     12   2111.601  0.0001
M1*AGE         2    0    398    0.99430214     0.7611      6       1596  0.6006
M2*AGE         2    0    398    0.98077754     2.5940      6       1596  0.0167
MD*AGE         2    0    398    0.99564495     0.5811      6       1596  0.7457
-----------------------------------------------------------------------------

Dependent Variable: LOGERR_0
                                     Sum of           Mean
Source                  DF          Squares         Square   F Value      Pr > F
Model                   63      215.2039574      3.4159358      1.59      0.0031
Error                  800     1716.8791598      2.1460989
Corrected Total        863     1932.0831172

                  R-Square             C.V.       Root MSE        LOGERR_0 Mean
                  0.111384         38.07561       1.464957             3.847495

Source                  DF         Anova SS    Mean Square   F Value      Pr > F
A1                       1       0.10895645     0.10895645      0.05      0.8218
A2                       1       9.56335372     9.56335372      4.46      0.0351
A1*A2                    1       1.39258188     1.39258188      0.65      0.4207
DSAMP                    2      25.04577945    12.52288972      5.84      0.0030
A1*DSAMP                 2       4.94195035     2.47097517      1.15      0.3167
A2*DSAMP                 2       3.69310876     1.84655438      0.86      0.4234
KB_B                     2      32.62199634    16.31099817      7.60      0.0005
A1*KB_B                  2      10.58600709     5.29300355      2.47      0.0855
A2*KB_B                  2       1.70114382     0.85057191      0.40      0.6729
```

```
DSAMP*KB_B               4      6.40258561     1.60064640      0.75     0.5609
M1                       1      2.24408473     2.24408473      1.05     0.3068
A1*M1                    1      3.51508367     3.51508367      1.64     0.2010
A2*M1                    1      3.83788365     3.83788365      1.79     0.1815
DSAMP*M1                 2      3.68130305     1.84065153      0.86     0.4245
KB_B*M1                  2      2.80580492     1.40290246      0.65     0.5204
M2                       1      3.35846545     3.35846545      1.56     0.2113
A1*M2                    1      0.00489509     0.00489509      0.00     0.9619
A2*M2                    1      6.72813183     6.72813183      3.14     0.0770
DSAMP*M2                 2      0.74461203     0.37230601      0.17     0.8408
KB_B*M2                  2      1.60224909     0.80112455      0.37     0.6886
M1*M2                    1      0.55817768     0.55817768      0.26     0.6102
MD                       1     13.62890342    13.62890342      6.35     0.0119
A1*MD                    1      0.67457806     0.67457806      0.31     0.5752
A2*MD                    1      0.02339449     0.02339449      0.01     0.9169
DSAMP*MD                 2      8.75011346     4.37505673      2.04     0.1309
KB_B*MD                  2      3.38773648     1.69386824      0.79     0.4545
M1*MD                    1      0.51384869     0.51384869      0.24     0.6247
M2*MD                    1      0.48663575     0.48663575      0.23     0.6341
AGE                      2      4.60942562     2.30471281      1.07     0.3422
A1*AGE                   2     20.55358391    10.27679196      4.79     0.0086
A2*AGE                   2      0.71709620     0.35854810      0.17     0.8462
DSAMP*AGE                4     17.09028066     4.27257017      1.99     0.0940
KB_B*AGE                 4     15.74669230     3.93667307      1.83     0.1202
M1*AGE                   2      3.29015373     1.64507687      0.77     0.4650
M2*AGE                   2      0.19089303     0.09544651      0.04     0.9565
MD*AGE                   2      0.40246689     0.20123344      0.09     0.9105
------------------------------------------------------------------------------

Dependent Variable: LOGERR_1
                                    Sum of           Mean
Source                  DF         Squares         Square   F Value     Pr > F
Model                   63     297.7470516      4.7261437      3.25     0.0001
Error                  800    1163.1356081      1.4539195
Corrected Total        863    1460.8826597

                  R-Square            C.V.       Root MSE        LOGERR_1 Mean
                  0.203813        30.29255       1.205786             3.980470

Source                  DF        Anova SS    Mean Square   F Value     Pr > F
A1                       1     58.19559376    58.19559376     40.03     0.0001
A2                       1      0.34139960     0.34139960      0.23     0.6281
A1*A2                    1      0.87708824     0.87708824      0.60     0.4376
DSAMP                    2     12.29030126     6.14515063      4.23     0.0149
A1*DSAMP                 2      2.38655760     1.19327880      0.82     0.4405
A2*DSAMP                 2      0.92607050     0.46303525      0.32     0.7274
KB_B                     2     79.87976209    39.93988104     27.47     0.0001
A1*KB_B                  2     12.97861242     6.48930621      4.46     0.0118
A2*KB_B                  2      2.63697584     1.31848792      0.91     0.4042
DSAMP*KB_B               4     11.11785123     2.77946281      1.91     0.1065
```

```
M1                       1      4.02047489     4.02047489      2.77      0.0967
A1*M1                    1      2.89242700     2.89242700      1.99      0.1588
A2*M1                    1      5.42320900     5.42320900      3.73      0.0538
DSAMP*M1                 2      3.02935326     1.51467663      1.04      0.3533
KB_B*M1                  2      0.04391145     0.02195572      0.02      0.9850
M2                       1      0.07266690     0.07266690      0.05      0.8232
A1*M2                    1      0.26905154     0.26905154      0.19      0.6672
A2*M2                    1      1.23558040     1.23558040      0.85      0.3569
DSAMP*M2                 2      0.59202604     0.29601302      0.20      0.8158
KB_B*M2                  2      2.50248564     1.25124282      0.86      0.4233
M1*M2                    1      9.09796481     9.09796481      6.26      0.0126
MD                       1      3.72231534     3.72231534      2.56      0.1100
A1*MD                    1      0.06192721     0.06192721      0.04      0.8365
A2*MD                    1      0.69779612     0.69779612      0.48      0.4886
DSAMP*MD                 2      2.13796847     1.06898424      0.74      0.4797
KB_B*MD                  2      0.06083425     0.03041713      0.02      0.9793
M1*MD                    1      0.47411965     0.47411965      0.33      0.5681
M2*MD                    1      2.11207902     2.11207902      1.45      0.2285
AGE                      2      8.38553969     4.19276984      2.88      0.0565
A1*AGE                   2     16.03608976     8.01804488      5.51      0.0042
A2*AGE                   2      3.85220216     1.92610108      1.32      0.2664
DSAMP*AGE                4      6.51860432     1.62965108      1.12      0.3453
KB_B*AGE                 4     38.23487536     9.55871884      6.57      0.0001
M1*AGE                   2      1.38613040     0.69306520      0.48      0.6210
M2*AGE                   2      0.88626012     0.44313006      0.30      0.7374
MD*AGE                   2      2.37094629     1.18547314      0.82      0.4428
------------------------------------------------------------------------------

Dependent Variable: LOGERR_2
                                    Sum of           Mean
Source                  DF         Squares         Square   F Value       Pr > F
Model                   63     431.7471444      6.8531293      6.90       0.0001
Error                  800     794.8552135      0.9935690
Corrected Total        863    1226.6023578

                  R-Square            C.V.       Root MSE        LOGERR_2 Mean
                  0.351986        26.47052       0.996779             3.765620

Source                  DF        Anova SS    Mean Square   F Value       Pr > F
A1                       1       6.3269019      6.3269019      6.37       0.0118
A2                       1       3.8545064      3.8545064      3.88       0.0492
A1*A2                    1       0.1504221      0.1504221      0.15       0.6973
DSAMP                    2      12.0299061      6.0149531      6.05       0.0025
A1*DSAMP                 2       0.3667016      0.1833508      0.18       0.8315
A2*DSAMP                 2       0.1877698      0.0938849      0.09       0.9098
KB_B                     2     321.1409057    160.5704528    161.61       0.0001
A1*KB_B                  2       5.3116228      2.6558114      2.67       0.0697
A2*KB_B                  2       2.6683499      1.3341750      1.34       0.2617
DSAMP*KB_B               4       6.3597718      1.5899430      1.60       0.1723
M1                       1       0.0019535      0.0019535      0.00       0.9646
```

```
A1*M1                   1       1.5666847      1.5666847      1.58     0.2096
A2*M1                   1       1.4588801      1.4588801      1.47     0.2260
DSAMP*M1                2       3.1029922      1.5514961      1.56     0.2105
KB_B*M1                 2       0.1232305      0.0616153      0.06     0.9399
M2                      1       1.6005609      1.6005609      1.61     0.2047
A1*M2                   1       0.0422242      0.0422242      0.04     0.8367
A2*M2                   1       2.5675394      2.5675394      2.58     0.1083
DSAMP*M2                2       0.1590781      0.0795391      0.08     0.9231
KB_B*M2                 2       2.9251403      1.4625701      1.47     0.2301
M1*M2                   1       0.0348202      0.0348202      0.04     0.8515
MD                      1       6.0406606      6.0406606      6.08     0.0139
A1*MD                   1       0.1843521      0.1843521      0.19     0.6668
A2*MD                   1       0.5950511      0.5950511      0.60     0.4392
DSAMP*MD                2       2.8842473      1.4421237      1.45     0.2348
KB_B*MD                 2       3.7630774      1.8815387      1.89     0.1512
M1*MD                   1       3.8028271      3.8028271      3.83     0.0508
M2*MD                   1       0.0422493      0.0422493      0.04     0.8367
AGE                     2       1.5378230      0.7689115      0.77     0.4616
A1*AGE                  2      14.2707567      7.1353783      7.18     0.0008
A2*AGE                  2       4.9439904      2.4719952      2.49     0.0837
DSAMP*AGE               4       5.3362993      1.3340748      1.34     0.2524
KB_B*AGE                4       5.6259939      1.4064985      1.42     0.2269
M1*AGE                  2       4.3484114      2.1742057      2.19     0.1128
M2*AGE                  2       5.6363075      2.8181538      2.84     0.0592
MD*AGE                  2       0.7551349      0.3775675      0.38     0.6840
-----------------------------------------------------------------------------
-----------------------------------------------------------------------------


KB_D=1000

MULTIVARIATE ANALYSIS OF VARIANCE

Source        S    M     N    Wilk's Lambda     F       df1       df2  Pr > F
A1            1   0.5   398    0.99394206     1.6212     3       798  0.1830
A2            1   0.5   398    0.99679857     0.8543     3       798  0.4645
A1*A2         1   0.5   398    0.99491437     1.3597     3       798  0.2539
DSAMP         2    0    398    0.99487683     0.6840     6      1596  0.6626
A1*DSAMP      2    0    398    0.99123078     1.1740     6      1596  0.3174
A2*DSAMP      2    0    398    0.98873097     1.5116     6      1596  0.1705
KB_B          2    0    398    0.41855136   145.1566     6      1596  0.0001
A1*KB_B       2    0    398    0.95368330     6.3827     6      1596  0.0001
A2*KB_B       2    0    398    0.97822169     2.9447     6      1596  0.0073
DSAMP*KB_B    3    0    398    0.96148471     2.6317    12  2111.601  0.0017
M1            1   0.5   398    0.99750601     0.6651     3       798  0.5737
A1*M1         1   0.5   398    0.99932408     0.1799     3       798  0.9100
A2*M1         1   0.5   398    0.99785741     0.5712     3       798  0.6341
DSAMP*M1      2    0    398    0.98800564     1.6097     6      1596  0.1406
KB_B*M1       2    0    398    0.99499991     0.6675     6      1596  0.6760
M2            1   0.5   398    0.99756025     0.6506     3       798  0.5827
```

```
A1*M2         1   0.5   398    0.99669338     0.8825       3       798  0.4497
A2*M2         1   0.5   398    0.99764906     0.6268       3       798  0.5978
DSAMP*M2      2    0    398    0.99703711     0.3949       6      1596  0.8826
KB_B*M2       2    0    398    0.99751565     0.3310       6      1596  0.9208
M1*M2         1   0.5   398    0.99759142     0.6422       3       798  0.5880
MD            1   0.5   398    0.99840178     0.4258       3       798  0.7346
A1*MD         1   0.5   398    0.99665074     0.8939       3       798  0.4438
A2*MD         1   0.5   398    0.99819790     0.4802       3       798  0.6961
DSAMP*MD      2    0    398    0.99012027     1.3238       6      1596  0.2430
KB_B*MD       2    0    398    0.98940564     1.4203       6      1596  0.2031
M1*MD         1   0.5   398    0.99477814     1.3963       3       798  0.2426
M2*MD         1   0.5   398    0.99592904     1.0873       3       798  0.3536
AGE           2    0    398    0.95194708     6.6310       6      1596  0.0001
A1*AGE        2    0    398    0.99149986     1.1378       6      1596  0.3378
A2*AGE        2    0    398    0.99433163     0.7571       6      1596  0.6038
DSAMP*AGE     3    0    398    0.98231216     1.1909      12  2111.601  0.2834
KB_B*AGE      3    0    398    0.86658944     9.7859      12  2111.601  0.0001
M1*AGE        2    0    398    0.99798695     0.2681       6      1596  0.9519
M2*AGE        2    0    398    0.99469656     0.7082       6      1596  0.6431
MD*AGE        2    0    398    0.99532550     0.6239       6      1596  0.7113
-----------------------------------------------------------------------------

Dependent Variable: LOGERR_0
                                        Sum of           Mean
Source                      DF         Squares         Square   F Value      Pr > F
Model                       63     215.3988431      3.4190293      1.70      0.0009
Error                      800    1611.0685830      2.0138357
Corrected Total            863    1826.4674261

                      R-Square            C.V.       Root MSE        LOGERR_0 Mean
                      0.117932        23.02700       1.419097             6.162752

Source                      DF        Anova SS    Mean Square   F Value      Pr > F
A1                           1      0.99107525     0.99107525      0.49      0.4832
A2                           1      3.13071693     3.13071693      1.55      0.2128
A1*A2                        1      7.63144796     7.63144796      3.79      0.0519
DSAMP                        2      4.29919439     2.14959720      1.07      0.3444
A1*DSAMP                     2      1.94234712     0.97117356      0.48      0.6176
A2*DSAMP                     2      1.75828831     0.87914415      0.44      0.6464
KB_B                         2     45.14810624    22.57405312     11.21      0.0001
A1*KB_B                      2     26.46782667    13.23391334      6.57      0.0015
A2*KB_B                      2     11.14790530     5.57395265      2.77      0.0634
DSAMP*KB_B                   4     23.17975115     5.79493779      2.88      0.0220
M1                           1      0.96233329     0.96233329      0.48      0.4896
A1*M1                        1      0.03073424     0.03073424      0.02      0.9017
A2*M1                        1      0.60304981     0.60304981      0.30      0.5844
DSAMP*M1                     2      7.94836793     3.97418396      1.97      0.1397
KB_B*M1                      2      1.78387002     0.89193501      0.44      0.6423
M2                           1      1.48496036     1.48496036      0.74      0.3908
A1*M2                        1      0.35740174     0.35740174      0.18      0.6737
```

```
A2*M2                    1      2.08899787     2.08899787      1.04     0.3088
DSAMP*M2                 2      1.48821105     0.74410552      0.37     0.6912
KB_B*M2                  2      0.59557870     0.29778935      0.15     0.8626
M1*M2                    1      0.74829307     0.74829307      0.37     0.5423
MD                       1      1.91561961     1.91561961      0.95     0.3297
A1*MD                    1      0.25327899     0.25327899      0.13     0.7230
A2*MD                    1      2.74550500     2.74550500      1.36     0.2433
DSAMP*MD                 2      4.80169492     2.40084746      1.19     0.3041
KB_B*MD                  2      0.54143885     0.27071943      0.13     0.8742
M1*MD                    1      1.31750614     1.31750614      0.65     0.4188
M2*MD                    1      0.00436671     0.00436671      0.00     0.9629
AGE                      2      8.13599683     4.06799841      2.02     0.1333
A1*AGE                   2      0.76691063     0.38345532      0.19     0.8267
A2*AGE                   2      7.12731882     3.56365941      1.77     0.1711
DSAMP*AGE                4     11.45879379     2.86469845      1.42     0.2246
KB_B*AGE                 4     27.07323883     6.76830971      3.36     0.0097
M1*AGE                   2      0.22944435     0.11472217      0.06     0.9446
M2*AGE                   2      3.94980721     1.97490360      0.98     0.3755
MD*AGE                   2      1.28946507     0.64473253      0.32     0.7261
------------------------------------------------------------------------------

Dependent Variable: LOGERR_1
                                    Sum of           Mean
Source                  DF         Squares         Square   F Value     Pr > F
Model                   63     215.4632380      3.4200514      1.92     0.0001
Error                  800    1424.5757862      1.7807197
Corrected Total        863    1640.0390242

                  R-Square            C.V.       Root MSE        LOGERR_1 Mean
                  0.131377        21.50321       1.334436             6.205753

Source                  DF        Anova SS    Mean Square   F Value     Pr > F
A1                       1      3.83907978     3.83907978      2.16     0.1424
A2                       1      1.08700143     1.08700143      0.61     0.4349
A1*A2                    1      5.89142027     5.89142027      3.31     0.0693
DSAMP                    2      5.97519619     2.98759810      1.68     0.1875
A1*DSAMP                 2      0.44131373     0.22065686      0.12     0.8835
A2*DSAMP                 2      4.01062194     2.00531097      1.13     0.3248
KB_B                     2     39.22985782    19.61492891     11.02     0.0001
A1*KB_B                  2     45.45870523    22.72935262     12.76     0.0001
A2*KB_B                  2      9.03755288     4.51877644      2.54     0.0797
DSAMP*KB_B               4     18.98069674     4.74517419      2.66     0.0314
M1                       1      0.77636692     0.77636692      0.44     0.5093
A1*M1                    1      0.32501498     0.32501498      0.18     0.6693
A2*M1                    1      0.92276178     0.92276178      0.52     0.4718
DSAMP*M1                 2      3.23249211     1.61624605      0.91     0.4039
KB_B*M1                  2      1.56380804     0.78190402      0.44     0.6448
M2                       1      2.75054087     2.75054087      1.54     0.2143
A1*M2                    1      0.00952186     0.00952186      0.01     0.9417
A2*M2                    1      1.40581767     1.40581767      0.79     0.3745
```

```
DSAMP*M2                 2      0.20077210     0.10038605      0.06     0.9452
KB_B*M2                  2      0.67664730     0.33832365      0.19     0.8270
M1*M2                    1      1.07455608     1.07455608      0.60     0.4375
MD                       1      0.78446439     0.78446439      0.44     0.5071
A1*MD                    1      0.60374345     0.60374345      0.34     0.5605
A2*MD                    1      1.99305379     1.99305379      1.12     0.2904
DSAMP*MD                 2      6.81970569     3.40985285      1.91     0.1480
KB_B*MD                  2      3.47032067     1.73516034      0.97     0.3779
M1*MD                    1      2.15003578     2.15003578      1.21     0.2722
M2*MD                    1      0.06244464     0.06244464      0.04     0.8515
AGE                      2     10.47714909     5.23857455      2.94     0.0533
A1*AGE                   2      0.14506910     0.07253455      0.04     0.9601
A2*AGE                   2      6.08755889     3.04377944      1.71     0.1817
DSAMP*AGE                4      9.16945078     2.29236270      1.29     0.2733
KB_B*AGE                 4     22.88525646     5.72131411      3.21     0.0125
M1*AGE                   2      0.64294511     0.32147255      0.18     0.8349
M2*AGE                   2      3.16121471     1.58060736      0.89     0.4120
MD*AGE                   2      0.12107974     0.06053987      0.03     0.9666
 KB_D=1000
Dependent Variable: LOGERR_2
                                    Sum of           Mean
Source                  DF         Squares         Square   F Value     Pr > F
Model                   63     555.9469381      8.8245546     13.97     0.0001
Error                  800     505.5201303      0.6319002
Corrected Total        863    1061.4670684

                  R-Square            C.V.       Root MSE        LOGERR_2 Mean
                  0.523753        12.59837       0.794921             6.309715

Source                  DF        Anova SS    Mean Square   F Value     Pr > F
A1                       1       0.0001852      0.0001852      0.00     0.9863
A2                       1       0.8561766      0.8561766      1.35     0.2448
A1*A2                    1       1.6305209      1.6305209      2.58     0.1086
DSAMP                    2       1.5676162      0.7838081      1.24     0.2898
A1*DSAMP                 2       1.2208569      0.6104284      0.97     0.3810
A2*DSAMP                 2       2.0265496      1.0132748      1.60     0.2018
KB_B                     2     481.3262841    240.6631421    380.86     0.0001
A1*KB_B                  2       0.1638387      0.0819193      0.13     0.8784
A2*KB_B                  2       1.2663695      0.6331847      1.00     0.3676
DSAMP*KB_B               4       2.4806631      0.6201658      0.98     0.4168
M1                       1       1.2359318      1.2359318      1.96     0.1623
A1*M1                    1       0.0163951      0.0163951      0.03     0.8721
A2*M1                    1       0.1313388      0.1313388      0.21     0.6486
DSAMP*M1                 2       2.7743708      1.3871854      2.20     0.1120
KB_B*M1                  2       2.0699653      1.0349827      1.64     0.1950
M2                       1       0.4334017      0.4334017      0.69     0.4078
A1*M2                    1       0.4026609      0.4026609      0.64     0.4250
A2*M2                    1       0.0088598      0.0088598      0.01     0.9058
DSAMP*M2                 2       0.1875613      0.0937807      0.15     0.8621
KB_B*M2                  2       0.6898768      0.3449384      0.55     0.5796
```

```
M1*M2                    1        0.1384554       0.1384554       0.22      0.6398
MD                       1        0.0560449       0.0560449       0.09      0.7659
A1*MD                    1        0.4983048       0.4983048       0.79      0.3748
A2*MD                    1        0.1465879       0.1465879       0.23      0.6302
DSAMP*MD                 2        1.2922843       0.6461421       1.02      0.3602
KB_B*MD                  2        0.0907604       0.0453802       0.07      0.9307
M1*MD                    1        0.3474409       0.3474409       0.55      0.4586
M2*MD                    1        1.2339226       1.2339226       1.95      0.1627
AGE                      2       14.6341266       7.3170633      11.58      0.0001
A1*AGE                   2        1.2896081       0.6448041       1.02      0.3609
A2*AGE                   2        2.2177744       1.1088872       1.75      0.1736
DSAMP*AGE                4        3.1780990       0.7945248       1.26      0.2853
KB_B*AGE                 4       27.5369938       6.8842485      10.89      0.0001
M1*AGE                   2        0.4668154       0.2334077       0.37      0.6913
M2*AGE                   2        1.3108894       0.6554447       1.04      0.3549
MD*AGE                   2        1.0194072       0.5097036       0.81      0.4467
------------------------------------------------------------------------------
```

C.1.2 Tables of Estimates

```
KB_D = -50

                                 No Smooth  Mutation Smooth  Restrictive Smooth
                              MSE = 0.0379     MSE = 0.0402        MSE = 0.0236

Factor Levels              N          Mean             Mean                Mean
-------------------------------------------------------------------------------
a1=2, a2=2, kb_b=20       72       4.05539          4.05808             4.05616
a1=2, a2=2, kb_b=200               3.99213          3.99042             3.99213
a1=2, a2=2, kb_b=2000              3.93183          3.93183             3.93183
a1=2, a2=8, kb_b=20                4.19112          4.26268             4.19398
a1=2, a2=8, kb_b=200               4.24473          4.46453             4.24343
a1=2, a2=8, kb_b=2000              3.93183          3.93183             3.93183
a1=6, a2=2, kb_b=20                3.93014          3.93183             3.93183
a1=6, a2=2, kb_b=200               3.92604          3.93183             3.93183
a1=6, a2=2, kb_b=2000              3.93431          3.93183             3.93183
a1=6, a2=8, kb_b=20                4.16886          4.23511             4.17997
a1=6, a2=8, kb_b=200               3.93183          3.93183             3.93183
a1=6, a2=8, kb_b=2000              3.93431          3.93183             3.93431
-------------------------------------------------------------------------------
a2=2, m2=0               216       3.97321          3.97415             3.96338
a2=2, m2=1e-4                      3.95007          3.95112             3.96182
a2=6, m2=0                         4.09832          4.12731             4.09789
a2=6, m2=1e-4                      4.03590          4.12529             4.04056
-------------------------------------------------------------------------------
dsamp=25,  m2=0          144       4.01671          4.02930             4.01606
dsamp=25,  m2=1e-4                 4.01368          4.06393             4.04107
dsamp=150, m2=0                    4.05035          4.06579             4.03561
dsamp=150, m2=1e-4                 4.00319          4.04430             3.99929
dsamp=250, m2=0                    4.04024          4.05710             4.04024
dsamp=250, m2=1e-4                 3.96209          4.00637             3.96320
-------------------------------------------------------------------------------
kb_b=20,   m2=0          144       4.13704          4.14201             4.13704
kb_b=20,   m2=1e-4                 4.03572          4.10183             4.04393
kb_b=200,  m2=0                    4.03595          4.07835             4.02179
kb_b=200,  m2=1e-4                 4.01142          4.08095             4.02781
kb_b=2000, m2=0                    3.93431          3.93183             3.93307
kb_b=2000, m2=1e-4                 3.93183          3.93183             3.93183
-------------------------------------------------------------------------------
a1=2, kb_b=20,   age=100  48       4.12962          4.17186             4.14497
a1=2, kb_b=20,   age=200           4.15026          4.15235             4.14037
a1=2, kb_b=20,   age=400           4.08988          4.15694             4.08988
a1=2, kb_b=200,  age=100           4.30747          4.29460             4.25250
a1=2, kb_b=200,  age=200           4.03021          4.22748             4.08324
a1=2, kb_b=200,  age=400           4.01760          4.16035             4.01760
a1=2, kb_b=2000, age=100           3.93183          3.93183             3.93183
a1=2, kb_b=2000, age=200           3.93183          3.93183             3.93183
```

```
a1=2, kb_b=2000, age=400         3.93183          3.93183             3.93183
a1=2, kb_b=20,   age=100         4.07509          4.08347             4.07657
a1=2, kb_b=20,   age=200         4.04954          4.08347             4.06279
a1=2, kb_b=20,   age=400         4.02387          4.08347             4.02832
a1=2, kb_b=200,  age=100         3.92398          3.93183             3.93183
a1=2, kb_b=200,  age=200         3.93183          3.93183             3.93183
a1=2, kb_b=200,  age=400         3.93099          3.93183             3.93183
a1=2, kb_b=2000, age=100         3.93556          3.93183             3.93556
a1=2, kb_b=2000, age=200         3.93556          3.93183             3.93183
a1=2, kb_b=2000, age=400         3.93183          3.93183             3.93183
-------------------------------------------------------------------------------
-------------------------------------------------------------------------------


KB_D = 10

                                 No Smooth  Mutation Smooth  Restrictive Smooth
                              MSE = 1.3329     MSE = 0.8090        MSE = 0.8196

Factor Levels              N          Mean             Mean                Mean
-------------------------------------------------------------------------------
a1=2, a2=2               216       2.29021          2.23339             1.99751
a1=2, a2=8                         2.47972          2.57300             2.34295
a1=6, a2=2                         2.30540          2.11634             2.06599
a1=6, a2=8                         2.13630          2.07921             1.99023
-------------------------------------------------------------------------------
a1=2, dsamp=25           144       2.55660          2.41989             2.39455
a1=2, dsamp=150                    2.40783          2.40745             2.07898
a1=2, dsamp=250                    2.19045          2.38225             2.03716
a1=6, dsamp=25                     2.74009          2.60554             2.40236
a1=6, dsamp=150                    2.15597          1.96566             1.92281
a1=6, dsamp=250                    1.76649          1.72213             1.75916
-------------------------------------------------------------------------------
a1=2, kb_b=20            144       2.12326          1.92841             1.66980
a1=2, kb_b=200                     2.72831          2.81932             2.56873
a1=2, kb_b=2000                    2.30332          2.46186             2.27215
a1=6, kb_b=20                      2.30036          2.10980             1.84011
a1=6, kb_b=200                     2.42277          2.32984             2.14710
a1=6, kb_b=2000                    1.93942          1.85369             2.09712
-------------------------------------------------------------------------------
a2=2, kb_b=20            144       2.14331          1.91115             1.67414
a2=2, kb_b=200                     2.48340          2.35372             2.17108
a2=2, kb_b=2000                    2.26670          2.25973             2.25002
a2=6, kb_b=20                      2.28030          2.12706             1.83577
a2=6, kb_b=200                     2.66768          2.79544             2.54475
a2=6, kb_b=2000                    1.97604          2.05582             2.11925
-------------------------------------------------------------------------------
dsamp=25,  kb_b=20        96       2.25936          2.05292             1.84042
dsamp=25,  kb_b=200                3.03564          2.96050             2.76729
dsamp=25,  kb_b=2000               2.65003          2.52472             2.58764
```

```
dsamp=150, kb_b=20               2.28595         2.09822           1.77206
dsamp=150, kb_b=200              2.46320         2.40023           2.24517
dsamp=150, kb_b=2000             2.09656         2.06121           1.98546
dsamp=250, kb_b=20               2.09010         1.90618           1.65239
dsamp=250, kb_b=200              2.22779         2.36300           2.06128
dsamp=250, kb_b=2000             1.61752         1.88739           1.98081
------------------------------------------------------------------------------
dsamp=25,  age=100        96     2.62632         2.54239           2.40477
dsamp=25,  age=200               2.78759         2.64751           2.51289
dsamp=25,  age=400               2.53112         2.34824           2.27770
dsamp=150, age=100               2.66479         2.40564           2.22233
dsamp=150, age=200               2.03880         1.95011           1.84586
dsamp=150, age=400               2.14212         2.20391           1.93449
dsamp=250, age=100               2.13554         2.15026           2.00022
dsamp=250, age=200               1.74702         1.88275           1.64284
dsamp=250, age=400               2.05285         2.12356           2.05143
------------------------------------------------------------------------------
kb_b=20,   age=100        96     2.40429         2.23099           1.98295
kb_b=20,   age=200               2.04112         1.77489           1.62498
kb_b=20,   age=400               2.19001         2.05144           1.65693
kb_b=200,  age=100               2.68863         2.57865           2.47665
kb_b=200,  age=200               2.50661         2.63129           2.25993
kb_b=200,  age=400               2.53138         2.51379           2.33716
kb_b=2000, age=100               2.33372         2.28865           2.16771
kb_b=2000, age=200               2.02569         2.07419           2.11667
kb_b=2000, age=400               2.00470         2.11048           2.26953
------------------------------------------------------------------------------
------------------------------------------------------------------------------



KB_D = 100

                                 No Smooth  Mutation Smooth  Restrictive Smooth
                              MSE = 2.1461     MSE = 1.4539        MSE = 0.9936

Factor Levels             N         Mean            Mean              Mean
------------------------------------------------------------------------------
a1=2, kb_b=20            144     4.11532         4.60081           4.47560
a1=2, kb_b=200                   3.45580         3.66329           3.08448
a1=2, kb_b=2000                  4.00506         4.45590           3.99351
a1=6, kb_b=20                    4.01921         3.99565           4.49910
a1=6, kb_b=200                   3.73369         3.47809           2.90804
a1=6, kb_b=2000                  3.75590         3.68908           3.63300
------------------------------------------------------------------------------
a1=2, age=20             144     3.79002         4.19009           3.74124
a1=2, age=200                    4.00582         4.28527           3.95817
a1=2, age=2000                   3.78034         4.24465           3.85418
a1=6, age=20                     4.11141         4.04691           3.86436
a1=6, age=200                    3.57893         3.50483           3.45507
```

```
a1=6, age=2000                     3.81846          3.61108             3.72071
-------------------------------------------------------------------------------
a2=2, age=20             144       4.09665          4.08942             3.78107
a2=2, age=200                      3.87658          3.86948             3.76566
a2=2, age=2000                     3.88487          4.04215             3.95051
a2=6, age=20                       3.80477          4.14758             3.82453
a2=6, age=200                      3.70817          3.92062             3.64758
a2=6, age=2000                     3.71392          3.81357             3.62437
-------------------------------------------------------------------------------
kb_b=20,   age=100        96       3.95506          4.24673             4.46962
kb_b=20,   age=200                 4.13001          4.38959             4.49350
kb_b=20,   age=400                 4.11672          4.25838             4.49892
kb_b=200,  age=100                 3.66937          3.57315             3.07875
kb_b=200,  age=200                 3.51896          3.69787             3.00369
kb_b=200,  age=400                 3.59589          3.44105             2.90634
kb_b=2000, age=100                 4.22770          4.53562             3.86003
kb_b=2000, age=200                 3.72816          3.59770             3.62267
kb_b=2000, age=400                 3.68558          4.08415             3.95707
-------------------------------------------------------------------------------
-------------------------------------------------------------------------------



KB_D = 1000

                                 No Smooth  Mutation Smooth  Restrictive Smooth
                              MSE = 2.0138     MSE = 1.7807        MSE = 0.6319

Factor Levels              N          Mean             Mean                Mean
-------------------------------------------------------------------------------
a1=2, kb_b=20            144       6.53815          6.63198             6.89777
a1=2, kb_b=200                     6.25596          6.42626             6.79105
a1=2, kb_b=2000                    5.79575          5.75899             5.23893
a1=6, kb_b=20                      6.43281          6.36685             6.89752
a1=6, kb_b=200                     5.77954          5.80871             6.75885
a1=6, kb_b=2000                    6.17430          6.24172             5.27416
-------------------------------------------------------------------------------
a2=2, kb_b=20            144       6.47741          6.43115             6.89804
a2=2, kb_b=200                     6.23801          6.29211             6.78359
a2=2, kb_b=2000                    5.95343          6.00041             5.34196
a2=6, kb_b=20                      6.49355          6.56768             6.89725
a2=6, kb_b=200                     5.79749          5.94286             6.76632
a2=6, kb_b=2000                    6.01663          6.00031             5.17113
-------------------------------------------------------------------------------
dsamp=25,  kb_b=20        96       6.65095          6.65903             6.89861
dsamp=25,  kb_b=200                6.29298          6.40119             6.78881
dsamp=25,  kb_b=2000               5.82681          5.89768             5.41087
dsamp=150, kb_b=20                 6.37341          6.37090             6.89720
dsamp=150, kb_b=200                6.00125          6.10117             6.76837
dsamp=150, kb_b=2000               5.88675          5.89528             5.12390
```

```
dsamp=250, kb_b=20                 6.43207          6.46831             6.89713
dsamp=250, kb_b=200                5.75902          5.85010             6.76768
dsamp=250, kb_b=2000               6.24152          6.20811             5.23487
-------------------------------------------------------------------------------
kb_b=20,   age=100        96       6.37732          6.34565             6.89694
kb_b=20,   age=200                 6.47730          6.50824             6.89868
kb_b=20,   age=400                 6.60181          6.64436             6.89732
kb_b=200,  age=100                 5.97744          5.99066             6.76827
kb_b=200,  age=200                 5.82043          5.93915             6.76991
kb_b=200,  age=400                 6.25539          6.42265             6.78668
kb_b=2000, age=100                 6.38871          6.31032             5.03886
kb_b=2000, age=200                 5.78314          5.75138             4.93645
kb_b=2000, age=400                 5.78323          5.93938             5.79434
-------------------------------------------------------------------------------
-------------------------------------------------------------------------------
```

C.2 Coverage Probability

C.2.1 Analysis of Variance Tables

```
NO SMOOTH

               MAXIMUM-LIKELIHOOD ANALYSIS-OF-VARIANCE TABLE

              Source                 DF   Chi-Square      Prob
              --------------------------------------------------
              INTERCEPT               1       686.33    0.0000
              A1                      1         6.02    0.0141
              A2                      1        16.84    0.0000
              DSAMP                   2      1773.59    0.0000
              KB_B                    2        71.03    0.0000
              M1                      1         0.39    0.5346
              M2                      1         0.36    0.5482
              MD                      1         0.21    0.6474
              AGE                     2        73.56    0.0000
              KB_D                    3      1662.42    0.0000
              A1*KB_D                 3        12.65    0.0055
              A2*KB_D                 3        19.40    0.0002
              DSAMP*KB_D              5       996.10    0.0000
              KB_B*KB_D               6        32.54    0.0000
              AGE*KB_D                6       102.38    0.0000
              M1*KB_D                 3         1.26    0.7390
              M2*KB_D                 3         1.94    0.5840
              MD*KB_D                 3         0.79    0.8511

              LIKELIHOOD RATIO     3409      2986.69    1.0000
```

```
MUTATION SMOOTH

               MAXIMUM-LIKELIHOOD ANALYSIS-OF-VARIANCE TABLE

              Source                 DF   Chi-Square      Prob
              --------------------------------------------------
              INTERCEPT               1        27.97    0.0000
              A1                      1        12.51    0.0004
              A2                      1        14.83    0.0001
              DSAMP                   2       135.60    0.0000
              KB_B                    2        69.80    0.0000
              M1                      1         0.03    0.8566
              M2                      1         0.08    0.7717
              MD                      1         0.48    0.4879
              AGE                     2        86.12    0.0000
              KB_D                    3       277.70    0.0000
              A1*KB_D                 3        35.39    0.0000
              A2*KB_D                     3        13.49    0.0037
              DSAMP*KB_D                  6        67.79    0.0000
              KB_B*KB_D                   6       105.11    0.0000
              AGE*KB_D                    6       114.86    0.0000
              M1*KB_D                     3         1.50    0.6826
              M2*KB_D                     3         0.68    0.8769
              MD*KB_D                     3         6.32    0.0970

              LIKELIHOOD RATIO         3408      2648.81    1.0000


RESTRICTIVE SMOOTH

                MAXIMUM-LIKELIHOOD ANALYSIS-OF-VARIANCE TABLE

              Source                     DF   Chi-Square      Prob
              --------------------------------------------------
              INTERCEPT                   1        20.72    0.0000
              A1                          1         2.37    0.1234
              A2                          1        28.81    0.0000
              DSAMP                       2       169.11    0.0000
              KB_B                        2        66.33    0.0000
              M1                          1         0.09    0.7684
              M2                          1         3.00    0.0833
              MD                          1         0.65    0.4195
              AGE                         2       213.60    0.0000
              KB_D                        3       337.31    0.0000
              A1*KB_D                     3        17.70    0.0005
              A2*KB_D                     3         9.29    0.0257
              DSAMP*KB_D                  4       164.17    0.0000
              KB_B*KB_D                   6        98.66    0.0000
              AGE*KB_D                    5       242.20    0.0000
              M1*KB_D                     3         1.68    0.6423
              M2*KB_D                     3         2.32    0.5078
              MD*KB_D                     3         0.88    0.8310

              LIKELIHOOD RATIO         3411      1781.74    1.0000
```

C.2.2 Tables of Estimates

		No Smooth	Mutation Smooth	Restrictive Smooth
Factor Levels	N	Coverage	Coverage	Coverage
a1=2, kb_d=-50	432	0.18056	0.21991	0.17361
a1=2, kb_d=10		0.81713	0.90509	0.95370
a1=2, kb_d=100		0.62037	0.51852	0.81481
a1=2, kb_d=1000		0.56250	0.51620	0.63194
a1=6, kb_d=-50		0.14352	0.15741	0.13889
a1=6, kb_d=10		0.83565	0.86806	0.95602
a1=6, kb_d=100		0.59028	0.59954	0.85417
a1=6, kb_d=1000		0.59491	0.57639	0.71991
a2=2, kb_d=-50	432	0.18056	0.19907	0.17361
a2=2, kb_d=10		0.82407	0.90278	0.96991
a2=2, kb_d=100		0.61343	0.56481	0.85185
a2=2, kb_d=1000		0.65278	0.63194	0.75000
a2=6, kb_d=-50		0.14352	0.17824	0.13889
a2=6, kb_d=10		0.82870	0.87037	0.93981
a2=6, kb_d=100		0.59722	0.55324	0.81713
a2=6, kb_d=1000		0.50463	0.46065	0.60185
dsamp=25, kb_d=-50	288	0.48264	0.55903	0.46528
dsamp=25, kb_d=10		0.92014	0.96181	1.00000
dsamp=25, kb_d=100		0.76389	0.75694	0.92361
dsamp=25, kb_d=1000		0.77083	0.75000	0.78472
dsamp=150, kb_d=-50		0.00347	0.00347	0.00347
dsamp=150, kb_d=10		0.78125	0.90625	0.94444
dsamp=150, kb_d=100		0.52431	0.49306	0.80208
dsamp=150, kb_d=1000		0.53819	0.48958	0.63194
dsamp=250, kb_d=-50		0.00000	0.00347	0.00000
dsamp=250, kb_d=10		0.77778	0.79167	0.92014
dsamp=250, kb_d=100		0.52778	0.42708	0.77778
dsamp=250, kb_d=1000		0.42708	0.39931	0.61111
kb_b=20, kb_d=-50	288	0.09375	0.11458	0.08681
kb_b=20, kb_d=10		0.78819	0.92014	0.97917
kb_b=20, kb_d=100		0.49306	0.33681	0.60764
kb_b=20, kb_d=1000		0.47222	0.43056	0.47569
kb_b=200, kb_d=-50		0.17361	0.17361	0.16319
kb_b=200, kb_d=10		0.81944	0.86806	0.93750
kb_b=200, kb_d=100		0.66319	0.70833	0.95833
kb_b=200, kb_d=1000		0.58333	0.52431	0.63889
kb_b=2000, kb_d=-50		0.21875	0.27778	0.21875
kb_b=2000, kb_d=10		0.87153	0.87153	0.94792
kb_b=2000, kb_d=100		0.65972	0.63194	0.93750
kb_b=2000, kb_d=1000		0.68056	0.68403	0.91319

```
-------------------------------------------------------------------------------
age=100,   kb_d=-50       288   0.30556          0.32986             0.30556
age=100,   kb_d=10              0.89931          0.96181             0.98611
age=100,   kb_d=100             0.66319          0.63194             0.85417
age=100,   kb_d=1000            0.48958          0.44792             0.75000
age=200,   kb_d=-50             0.17014          0.17361             0.16319
age=200,   kb_d=10              0.81250          0.88889             0.95139
age=200,   kb_d=100             0.62500          0.59375             0.85417
age=200,   kb_d=1000            0.62500          0.61111             0.67361
age=4000,  kb_d=-50             0.01042          0.06250             0.00000
age=4000,  kb_d=10              0.76736          0.80903             0.92708
age=4000,  kb_d=100             0.52778          0.45139             0.79514
age=4000,  kb_d=1000            0.62153          0.57986             0.60417
-------------------------------------------------------------------------------
kb_d=-50                  864   0.16204          0.18866             0.15625
kb_d=10                         0.82639          0.88657             0.95486
kb_d=100                        0.60532          0.55903             0.83449
kb_d=1000                       0.57870          0.54630             0.67593
-------------------------------------------------------------------------------
-------------------------------------------------------------------------------
```