2013

Improving Inference in Population Genetics Using Statistics

Michal Palczewski

THE FLORIDA STATE UNIVERSITY

COLLEGE OF ARTS AND SCIENCES

IMPROVING INFERENCE IN POPULATION GENETICS USING STATISTICS

By

MICHAL PALCZEWSKI

A Dissertation submitted to the
Department of Scientific Computing
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Degree Awarded:
Spring Semester, 2013

Michal Palczewski defended this dissertation on March 26, 2013.

The members of the supervisory committee were:

Peter Beerli
Professor Directing Thesis

Anuj Srivastava
University Representative

Gordon Erlebacher
Committee Member

Alan Lemmon
Committee Member

Dennis Slice
Committee Member

The Graduate School has verified and approved the above-named committee members, and certifies that the dissertation has been approved in accordance with the university requirements.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

My studies at Florida State University focused on using computers and statistics to solve problems in population genetics. I have created models and algorithms that have the potential to improve the statistical analysis of population genetics. Population genetical data is often noisy and thus requires the use of statistics in order to be able to draw meaning from the data. This dissertation consists of three main projects. The first project involves the parallel evaluation an model inference on multi-locus data sets. Bayes factors are used for model selection. We used thermodynamic integration to calculate these Bayes factors. To be able to take advantage of parallel processing and parallelize calculation across a high performance computer cluster, I developed a new method to split the Bayes factor calculation into independent units and then combine them later. The next project, the Transition Probability Structured Coalescence [TSPC], involved the creation of a continuous approximation to the discrete migration process used in the structured coalescent that is commonly used to infer migration rates in biological populations. Previous methods required the simulation of these migration events, but there is little power to estimate the time and occurrence of these events. In my method, they are replaced with a one dimensional numerical integration. The third project involved the development of a model for the inference of the time of speciation. Previous models used a set time to delineate a speciation and speciation was a point process. Instead, this point process is replaced with a parameterized speciation model where each lineage speciates according to a parameterized distribution. This is effectively a broader model that allows both very quick and slow speciation. It also includes the previous model as a limiting case. These three project, although rather independent of each other, improve the inference of population genetic models and thus allow better analyses of genetic data in fields such as phylogeography, conservation, and epidemiology.

CHAPTER 1

INTRODUCTION

My studies at Florida State University focused on using computers and statistics to solve problems in population genetics. I have created models and algorithms that have the potential to improve the statistical analysis of population genetics. This first chapter will introduce the field of research and present a high level overview of my contribution. Chapter 2 will serve to give background information and introduce concepts that are used in later chapters. Chapters 3-5 describe the methods and algorithms in detail. Chapter 6 will finish with concluding remarks.

1.1 Population Genetics

Population genetics is the study of genetic change. Typically this applies to the individuals of one species. Though in a later chapter I begin to merge population genetics with speciation concepts.

The two main forces that act on any freely breeding population are genetic drift and mutation. Whereas mutation typically introduces genetic variety, genetic drift always reduces it. Genetic drift is a result of a population having a finite population size. The population size being finite, any two individuals in the same population are likely to have a common ancestor if one looks far enough back in the past. Biologists are often interested in this balance, as it is responsible for the genetic diversity of a species and tends to be a function of how many individuals there are. One high profile study that used population genetic methods was a study by Roman and Palumbi (2003), which estimated a lower bound on the estimate of the number of whales before whaling shrank their populations. This study showed that previous methods which relied on logs from whaling ships had greatly underestimated the population

size.

A single species is generally thought to be freely interbreeding. Often times however there is structure in the genetic process. For example: two groups of fish living in nearby lakes. Each set of fish is much more likely to interbreed with the fish in the same lake. If there is a stream connecting these lakes, then these fish may occasionally migrate from one lake to another and breed with the other lakes fish. If one lake is bigger than another, then there may be more fish in the larger lake. If one lake is upstream while the other downstream, then the migratory patterns between these lakes are likely to be asymmetric. Biologists are often interested in these migratory patterns. For example, (Jue et al., in prep) wanted to find where the grouper in the Gulf of Mexico is migrating. Bedford et al. (2010) investigated the patterns of dispersal of the common influenza virus, H3N2.

In Chapter 2, population genetics is covered in detail including the mathematical descriptions of the models that I have used or extended in my research.

1.2 Statistical Methods

Population genetical data are often noisy and thus require the use of statistics in order to be able to draw meaning from the data. I will be reviewing those statistical techniques that were of great impact on my research.

1.2.1 Bayesian Inference

The primary statistical method I have used is Bayesian inference. Bayesian inference is based on Bayes' rule, which allows us to make probability statements about the parameters we are interested in. This is in contrast to frequentist methods which typically can give us the degree of support our data have. For example, when inferring a population size. A Bayesian answer might be,"The population size has a 95% probability of being between 100 and 200". A more classical method would typically say "If we repeated our experiment many times, our inferred population size would be between 100 and 200, 95% of the time".

I have chosen to use Bayesian methods, not because I prefer one philosophy over another. I view both methods as valid and will use a likelihood analysis when the mathematics supports it, and I will use a Bayesian method when it seems more appropriate.

The most controversial part of Bayesian inference is the use of priors. Bayesian analysis requires prior distributions for parameters under investigation. These represent the belief for each outcome before running an experiment. Often uninformative priors are used to indicate that we do not have a biased belief. Often times a uniform distribution is used as an uninformative prior. However, even uniform priors have their limitations, which can lead to bias. They are characterized by a lower bound and an upper bound. Making these too wide or too narrow can effect the result of an analysis (Felsenstein, 2004).

Despite the shortfalls Bayesian inference continues to be a growing field, owing much to it's desirable mathematical qualities.

1.2.2 Markov Chain Monte Carlo (MCMC)

Markov Chain Monte Carlo is a stochastic sampling method. It allows for sampling from a probability distribution which we may not be able to calculate. This is very handy for Bayesian Inference. The solution to a problem when using Bayesian inference is the posterior distribution. When this distribution is known, the problem is solved and presenting the distribution is typically the answer.

After using MCMC to sample from an unknown probability distribution it is possible to estimate the density. This is typically done by creating a histogram. If these histograms look "rough" often a smoothing function will be used to make them seem smother.

1.3 Projects

1.3.1 Bayes Factors

The first project that I completed while at FSU was calculating Bayes factors using the program migrate. I did this in collaboration with Peter Beerli and it resulted in two publications Beerli and Palczewski (2010); Palczewski and Beerli (2012).

I was responsible for the bulk of the mathematics involved in using and computing the thermodynamic integration in our program. In addition, I have created a formula that allows us to parallelize our analysis across a computer cluster and then combine the results of multiple parallel analyses.

Bayes factors are a Bayesian method for model selection. Often someone may

desire to know whether or not two populations are in fact two population or just one large population, or whether migration is symmetric or asymmetric. With the population size or strength of migration having a smaller weight.

This project has allowed the calculation of Bayes factors for model selection in population genetics. At latest count, as of March 2013, our 2010 paper has been cited 66 times.

1.3.2 TPSC

The current method of inferring migration between population involves simulating many discrete migration events between two species. TPSC is an attempt to use a continuous approximation to this discrete process. The goal was to speed up inference and to allow the inference of large migration rates.

This project required creating a new model, and writing a computer program capable of Bayesian inference using MCMC. In addition during the creation of this process methods novel to population genetics were used to speed up inference, such as adaptive MCMC.

1.3.3 New Order Speciation

Current methods of speciation, typically use models that have a a discrete speciation time. In other words, at one point in time all individuals are on one species and then immediately they segregate and become two separate species. These models have had success Hey (2010).

However, these models are limited. It is believed that there are many cases like sympatric speciation(speciation that occurs without any physical distance or barrier), speciation can occur slowly. Current methods have no way of distinguishing between slow and fast migration.

Here, we propose and test a new model of speciation. These models allow each lineage to cross over from one species to another according to a parameterized distribution. This allows us to say something not just about when a speciation happens but also for how long.

CHAPTER 2

BACKGROUND

2.1 Introduction

Population Genetics is a mixture of statistics, biology, mathematics and computation. This is a brief summary of the methods that were used in my research. This is not meant as an exhaustive list rather it is meant to give the reader an overview of the concepts that will be used later.

2.1.1 The Felsenstein equation

No mathematical formula summarizes the interrelationship between data, trees, and parameters, better than the following equation.

$$L(\theta) = P(D|\theta) = \int_G P(G|\theta)P(D|G)dG \tag{2.1}$$

This is a high level equation and the details will be discussed through out the chapter. This was first published by Felsenstein (1988) and dubbed the Felsenstein equation by Hey and Nielsen (2007). θ is a vector containing all the population parameters in the model and the left side is the likelihood of those parameters. G represents a genealogy, otherwise known as a tree.

Kuhner et al. (1995) were able to compute this equation using integration by the Metropolis-Hastings algorithm. Further refinement was done by Beerli (1998) allowing for population structure and integrating with Markov Chain Monte Carlo over possible migrations. Wilson and Balding (1998) and Beaumont (1999) later stated this as a Bayesian problem. If we assume that our parameters, θ, come from

some prior distribution the problem can be stated by Bayes' theorem as

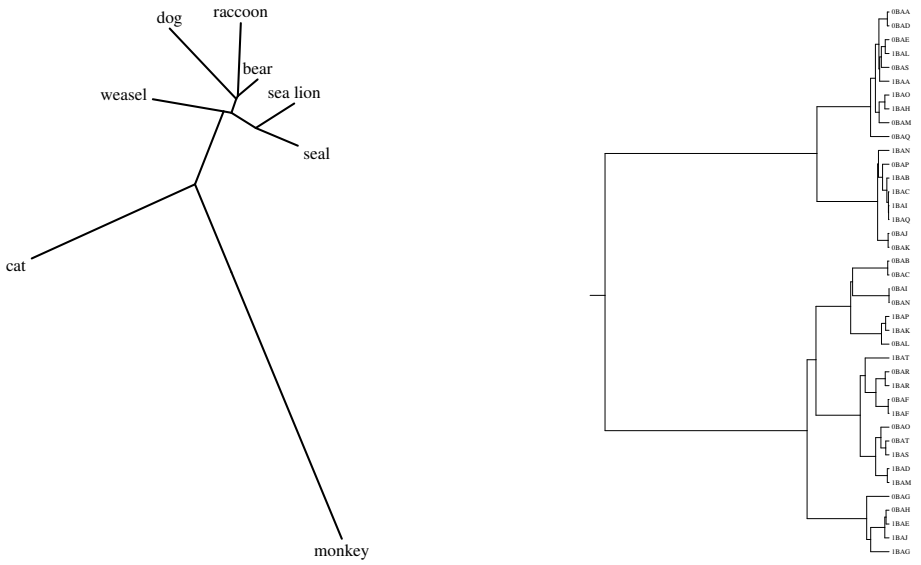
$$P(\theta, G|D) = \frac{P(\theta)P(D, G|\theta)}{P(D)} = \frac{P(\theta)P(G|\theta)P(D|G)}{P(D)} \tag{2.2}$$

One can sample from this distribution using the Markov Chain Monte Carlo(Section 2.6). The left side of this equation implies that both the population parameters and genealogies are sampled. However, since the genealogies are of no particular interest they are typically discarded in the final analysis.

2.2 Trees, Genealogies, Labeled Histories and Phylogenies

Trees are critical to the understanding of biology. Biology hinges on the theory of evolution, that claims that any two separate pieces of DNA have a common ancestor. DNA replicates by making a copy of itself, thus every relationship between individuals of wildly different species or of the same species can be represented as a binary tree.

Tree structures are theoretical constructs used in both Computer Science and Biology. Even though there is much overlap in the theory of how trees work between the two fields, the nomenclature is different. In computer science a tree is an acyclic connected graph. The terms "acyclic", "connected", and "graph" are very specific. A graph is a structure that contains nodes also known as vertices. These nodes may or may not have connections between them. These connections are known as branches or edges. Acyclic means that there are no loops in the graph structure. Connected means that one can start at any node, then by following branches arrive at any other node. These are the minimum properties of a tree, however, there are additional properties that a tree can have. A tree can be rooted or unrooted. In computer science this is known as directional or non-directional, respectively. A rooted tree has a root, which is a node with no parents representing a common ancestor. An unrooted tree does not show a common ancestor or any directionality. Instead it shows relationships(e.g. A and B are more closely related to each other than to C and D). A tree can have branch lengths. A node with only one branch is known as a tip in biology or a leaf in computer science. It usually represents a species or an individual sample. Two individuals in a tree represented by tips are considered to be more distantly related if the sum of the branch lengths between them is long rather



A: An unrooted tree B: A clock like rooted tree

Figure 2.1: Examples of rooted and unrooted trees. These trees were drawn with the programs drawtree and drawgram, respectively (Phylip 3.69 Felsenstein (1989)). The tree on the left was taken from an example in the Phylip documentation. The tree on the right was simulated.

then if the branch length was short. Likewise, a root that has a long branches leading out of it, indicates that the common ancestor in that tree was a very long time ago. Typically in biology, trees are bifurcating, meaning that an internal node in a tree has at most three branches leading in and out of it. Since trees represent the replication of a molecule, which is bifurcating in nature, non-bifurcating trees are considered to be unresolved.

Phylogenetic trees also known as phylogenies are trees that represent the interrelation of different species. Often these trees are inferred as unrooted trees. Figure 2.1A shows an unrooted tree.

Rooted clock like trees are the ones typically used in population genetic studies. The rates of evolution are assumed to be the same within a species. Clock like means that the trees all have tips that are at the same time. Figure 2.1B shows an example of such a tree.

To understand the framework in this dissertation, I need to introduce the concept

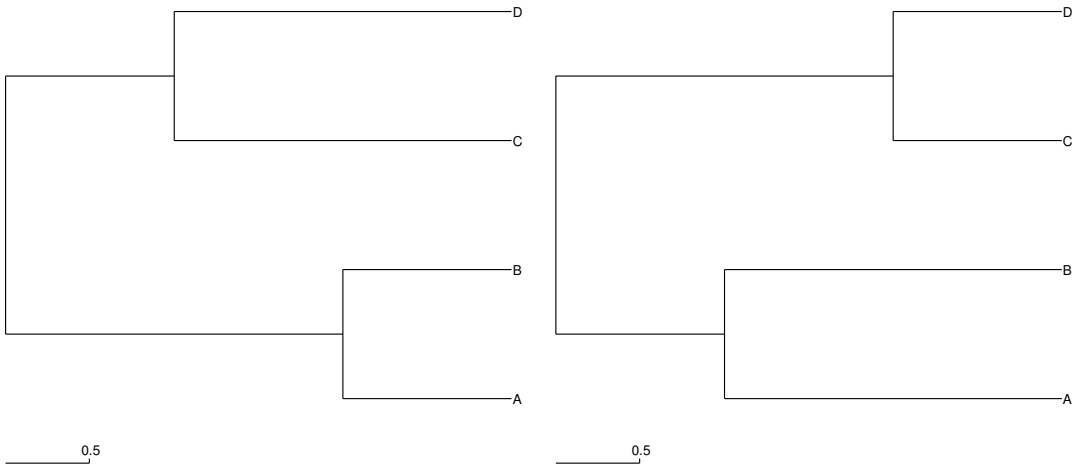


Figure 2.2: Trees versus labeled histories: These trees have the same topologies. When viewed as labeled histories, they are different because even though the topology is identical the order of the branching times is not.

Table 2.1: The number of labeled histories and rooted trees for a given amount of tips.

Number of tips	Number of rooted trees	Number of labeled histories
3	3	3
4	15	18
5	105	180
10	3.4459×10^{7}	2.5719×10^{9}
15	2.1346×10^{14}	$6.9581 \times^{18}$
20	8.2008×10^{21}	5.6448×10^{29}

of labeled histories. At first glance these appear to be the same as clock like rooted trees. The major difference is that the branching order of the times matter. Two labeled histories are said to have the same topology if the order of all the branching times are the same. This is demonstrated in Figure 2.2. When assuming a coalescent model(Section 2.3.1), each labeled history is just as likely as any other.

The number of possible trees grows very rapidly with the number of tips. Table 2.1 shows the number of possible trees for a given number of tips. It would be impossible to search every tree to find the best tree or to do an integral over all trees in an exhaustive manner. Typically for all but the smallest cases, heuristics are used to find the best tree. Other times, trees are a nuisance parameter in model. In these cases, probabilistic algorithms such as MCMC are used to integrate out all possible

trees, thus allowing an analysis of the parameters.

2.3 Population Genetics

Biologists are often interested in the population structure of the species they are studying. There may be a few quantitative parameters about the population of interest that they seek information about such as: population size, migration rates between populations, selection, recombination rates, growth rate, and etc.

There are two ways to estimate population parameters. The first is using direct observations of populations. The second is the use of indirect methods that use genetic data from individuals sampled from populations of interest (Beerli, 1998). I focus on these indirect methods of inference.

Wright and Fisher's population model (Wright, 1931; Fisher, 1930) is the prototypical model for population genetics. In this model there exist N individuals. Each one of these individuals has one copy of a gene. In the next generation the next group of individuals is formed by randomly choosing each gene with replacement from the previous generation. This model makes several assumptions: There is a fixed population size N, mating is random with respect to the gene being studied, generations do not overlap, there is no natural selection with respect to the gene being studied, and there is no migration from one population to another(i.e. there is no population structure.) Figure 2.3 shows one simulation of the Wright-Fisher model. Populations of diploid (each individual has two copies of a gene.) organisms are simply modeled as $2N$ gene copies. The two copies within an individual are independent.

Assuming a diploid population, the probability that any two genes share a common parent is $\frac{1}{2N}$ (This is slightly unrealistic for humans because two genes in one individual will not share the same parent.) The probability that two genes have the same ancestor 2 generations ago is equal to the probability that they do not share a common ancestor one generation ago times the probability that they share a common ancestor one more generation back, $\frac{1}{2N}\left(1-\frac{1}{2N}\right)$. This process can be repeated up to an arbitrary number of generations back in time. The general equation is

$$P(t_{mrca}) = \frac{1}{2N}\left(1-\frac{1}{2N}\right)^{t_{mrca}-1}. \tag{2.3}$$

Here t_{mrca} is the time until the most recent common ancestor. This is a geometric

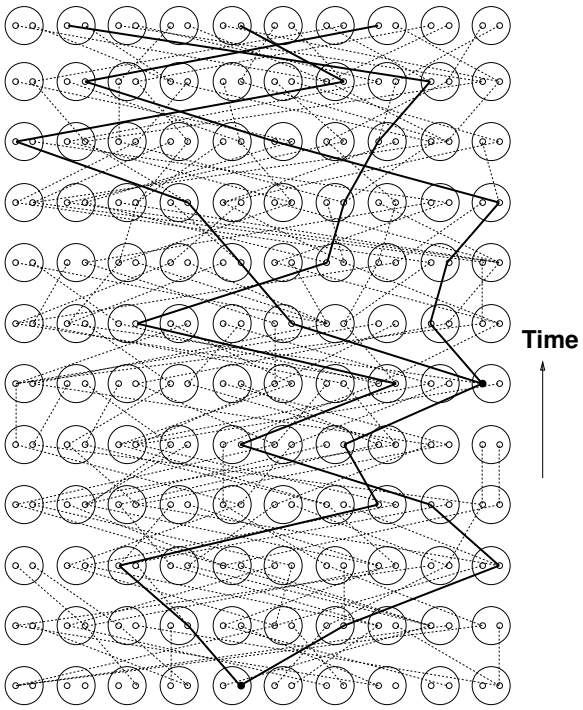


Figure 2.3: A simulated Wright-Fischer population. The dark lines show a possible sampling of lineages and their related genealogy. Image from, Felsenstein (1978b)

distribution. The expected value of a geometric distribution is the reciprocal of the probability of an event happening in one time step. Thus the expected time until a common ancestor is $2N$ generations for diploids.

2.3.1 The Coalescent

If the population size is large enough this geometric distribution can be approximated by an exponential distribution.

$$P(t_{mrca}) = \frac{1}{2N} e^{-\frac{1}{2N} t_{mrca}} \tag{2.4}$$

This formula is also the end result of taking a limit of letting the population size go to infinity while the generation time goes to zero. This process is known as the coalescent.

When dealing with more than two individuals, it is convenient to assume that the number of individuals of interest (k) is much smaller then N. Thus we ignore the very small probability that three individuals have the same parent. This assumption is very robust to violation and since it is based on the fact that the particles that create the genes only split one at a time (Wakeley, 2008).

Combinatorially there are $\binom{3}{2}$ ways for 3 particles to coalesce into two particles: particle 1 and 2, 2 and 3, or 3 and 1. When there are 4 or more individuals there are $\binom{4}{2}$. In general there will be $\binom{k}{2} = \frac{k(k-1)}{2}$ different ways that two individuals coalesce. So the probability that two lineages out of k have the same ancestor in the last generation is.

$$\frac{k(k-1)}{2} \frac{1}{2N} \tag{2.5}$$

Here k is the number of individuals that we have sampled or are interested in, while N is the entire population size. If formula 2.5 is a sufficiently small quantity, meaning that k is much smaller than N and N is large, then we can ignore the effect of three lineages sharing a common ancestor at the same time. The geometric distribution can then be approximated by an exponential distribution with the same mean.

In order to simulate the coalescent backwards in time, first draw a random interval of time from the exponential distribution describing the time until the first two sequences coalesce. Choose two random lineages to coalesce. Now the number of lineages is reduced by one and the process can be repeated until there is only one

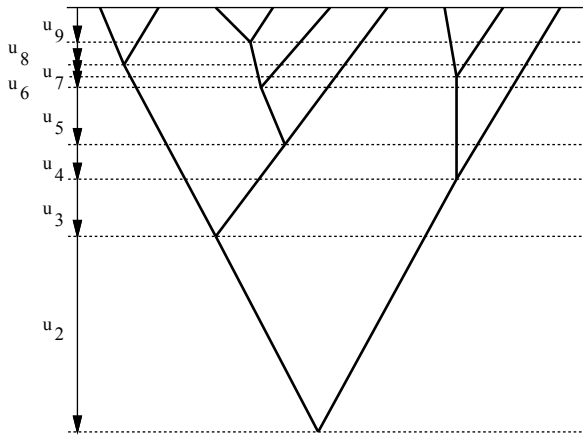


Figure 2.4: A genealogy simulated using the coalescent. The $u_1, u_2...$ represent the time until each individual coalescence. (Image from Felsenstein (1978b)

lineage left. The end result will form a bifurcating tree. This framework also allows us to calculate the probability of such a tree given a population size.

Given such a tree, first construct a time list $\{u_1, u_2, u_3, ...\}$. Where each u_i represents the time from the $(i-1)$ coalescence until the i coalescence. With u_1 simply equal to the time until the first coalescence. The probability of each time is stated.

$$P(u_i) = \frac{k(k-1)}{4N} e^{-\frac{k(k-1)}{4N} u_i}, \tag{2.6}$$

where time is measured in generations. That is only the probability density that at time u_1 an event occurred. On a tree, however, a specific event occurred at time t with the probability $\frac{2}{k(k-1)}$. Since each event on the tree occurs independently of other events the total tree probability is

$$P(G|N) = \prod_{i=1}^{k-1} \frac{2}{k(k-1)} \frac{k(k-1)}{4N} e^{-\frac{k(k-1)}{4N} u_i} = \prod_{i=1}^{k-1} \frac{2}{4N} e^{-\frac{k(k-1)}{4N} u_i}. \tag{2.7}$$

G represents the genealogy, otherwise known as the tree. I will continue to use these interchangeably as their meaning is the same, but the notation will change depending on which set of literature one is looking at. So far there has been one random mating population without selection or migration. This model will now be extended to allow for population structure. The coalescent was first extended to multiple populations

by Strobeck (1987).

Previously, simulating backwards in time required simulating an exponentially distributed interval of time, at which time a coalescence event would occur. If we allow for population structure then we assign every lineage into a population. Each population will have it's own rate of coalescence dependent on each population size. In addition to coalescent events, migration events can occur.

2.4 Migration

The Wright-Fisher model can be extended for multiple migrating populations. Instead of assuming that there is one population of randomly mating individuals one instead assumes that there are multiple populations. Before reproduction a random number of individuals migrate from population to another. This is governed by a migration rate m_{ij} that represents the rate from population i to j and is expressed in terms of the proportion of new offspring in population j that will have a parent from population i.

2.4.1 Coalescent and migration

Strobeck (1987) extended the coalescent to simulate genetic data coming from multiple populations. Later, Takahata (1988) showed the distribution of coalescences of two migrating lineages. This had the limitations of only allowing two lineages, assumed equal migration rates, and equal population sizes. In contrast to Strobeck, Takahata was able to overlay both the migration and coalescent processes on top of each other to create trees without implicit migration events. However, his method was slow and required the numerical computation of a matrix exponential(more on matrix exponential in 2.6) of a large matrix. The size of this matrix was equal to the number of samples. Allowing for a greater number of populations or migration rates would have further complicated the matrix exponential.

Beerli (1998) created a likelihood based method to infer migration rates and population sizes of multiple populations. This method can have a different value for each migration rate. Using this method for a multi population case, the rate λ at which

either a coalescent or migration events happen will be

$$\lambda_i = \sum_{i=j}^{P} \frac{k_j^i(k_j^i - 1)}{4N_j} + \sum_{i=1}^{P} \sum_{j=1, j\neq i}^{P} k_j m_{ij} \tag{2.8}$$

Where P is the total number of populations and k_i^j is the number of lineages currently in population i corresponding to the time before event i. At the begining of a simulation, lineages belong to the population in which they were sampled. Continuing the simulation backward in time, these lineages coalesce and migrate. m_{ij} is a migration rate defined as the percentage of the ancestors for the new generation of population j that were previously in i. λ_i is the rate at which the i'th event happens. Thus the probability of a tree given X number of events on the tree is

$$P(G|N, M) = \prod_{i=1}^{X} \frac{\beta_i}{\lambda_i} \lambda_i e^{\lambda_i} = \prod_{i-1}^{X} \beta_i e^{-\lambda_i t} \tag{2.9}$$

Here β_i is the contribution of the current event to the sum that makes lambda. This would be $\frac{2}{4N_i}$ for a coalescent event and m_{ij} for a migration event.

2.4.2 Violations of the Coalescent Model

In all these instances we have assumed an ideal Wright-Fisher population. Often it may not be the case that mating is random and binomially distributed. In these cases, it is enough to replace N, the population size, with N_e the effective population size. N_e is "the number of breeding individuals in an idealized population that would show the same amount of dispersion of allele frequencies under random genetic drift or the same amount of inbreeding as the population under consideration" (Wright, 1931, 1938). This is not a problem unique to the coalescent approximation. Furthermore, N_e and $\theta = 4N_e\mu$ (where μ is the mutation rate) are often of greater interest because they represent the amount of genetic variation.

It is quite difficult and sometimes impossible to untangle mutation rate from population size, because a low population size or a low mutation rate can lead to low diversity. Genealogies are usually estimated from genetic data. The lengths for the branches of estimated genealogies will be in units of expected mutations. Instead of N_e, the mutation-scaled effective population size $\theta = 4N_e\mu$ is estimated. Here μ is the mutation rate per generation. Since all tree lengths are in terms of expected number

```
0BAA  GGACGGCTTC
0BAB  GGGTAGCTTC
0BAC  AGGTAGGAAC
0BAD  AGATAGTACC
0BAE  ACATATTGCC
```

Figure 2.5: An example of five simulated DNA sequences. The first column are labels for each individual.

of mutations, the inference of migration is also effected: $M = m/\mu$ is estimated; m is the same as defined earlier for equation 2.8. Another useful term is $4N_e m$. This is the number of migrants per generation. When this value is higher than 1 different populations tend to act more like one large population and when it is less than 1 populations become more divergent.

2.5 Evolution models

The most common, data used to infer evolutionary trees such as phylogenies and genealogies are morphological characters, DNA or RNA sequence data, microsatellites, single nucleotide polymorphisms(SNP's), and restriction sites. Here I will focus on molecular sequence data from DNA or RNA. DNA is a long molecule made up smaller repeating molecules called bases. There are four bases: A, C, G and T(U for RNA). All our genetic information is held in DNA, while RNA is a messenger molecule that guides the expression of DNA. These can be sequenced and represented as long strings of characters. A small example of simulated DNA sequence data is shown in Figure 2.5. Typically these are much longer, however the length depends on the sequencing technology. On the short end as few as 200-500 base pairs are available. Newer longer runs will have many different loci of 100,000 base pairs each. These bases can be classified into purines (A or G) and pyrimidines (C or T/U). A particular organism can have multiple copies of the same chromosome referred to as ploidy. Haploid organisms have one copy of each gene whereas diploid organisms have two. Many organisms are diploid, such as humans, most animals, most plants and even some viruses.

Sometimes during replication mistakes called mutations will happen when DNA is copied. These mutations lead individuals to have different DNA sequences from their parents. Mutations happen frequently enough that sequences will be a different from

each other, but they are rare enough that much of the information on the relationship and genealogies of individuals remains. There are four common methods for inferring trees. Distance methods described by Fitch and Margoliash (1967), compute the distance that each sequence is from another and then find the tree which best fits the observed distances. Parsimony methods, first popularized by (Camin and Sokal, 1965), choose the tree that minimizes the number of mutations required to explain the tree. These methods are quick but can often lead to the wrong conclusion (Felsenstein, 1978a). Maximum likelihood estimates, first proposed by Felsenstein (1981), are implemented in Phylip (Felsenstein, 1989) and PAUP* (Swofford, 2003). Likelihood methods use a statistical model of evolution to calculate the probability of the data given the model and the tree. Bayesian methods, first implemented by Rannala and Yang (1996) to infer trees, put a prior on trees and use Bayes rule to calculate their posterior distribution. Aside from priors, Bayesian methods use the same statistical models as likelihood models.

2.5.1 Molecular Models

In order to sample genealogies, I will be using models of nucleotide evolution for inference. A comprehensive overview of all molecular models is outside the scope of this dissertation as entire books have been written about them (Felsenstein, 2004). However, I will briefly summarize them because I have used them in inference and to give the reader an idea of the scope of the problem.

The first model of nucleotide evolution was by Jukes and Cantor (1969). The assumptions were that each DNA or RNA site evolves independently of any other, mutations happen according to a Poisson process, every base appears equally as often as every other base in a stationary distribution, and every mutation happens at the same rate. Kimura (1980) extended this model to allow different rates of transitions and transversions. A transition is a mutation that transforms a purine(A or G) to the other purine or a pyrimidine(C or T) to the other pyrimidine. While a transversion transforms a purine into a pyrimidine or vice versa. Both the purines are much larger than the pyrimidines. A mutation that preserves the molecular size of the mutated molecule happens more frequently. Thus even though there are twice as many possible transversions as transitions they are generally observed to be more rare.

Felsenstein (1981) refined the Jukes-Cantor model in a different way. His model assumes that there are different equilibrium frequencies for each of the bases. This

means that if we simulated a single base for an infinite time there would be different probabilities of ending up at each nucleotide. The rate of each type of mutation depends only on these equilibrium frequencies. Felsenstein further showed how to calculate the probability of sequence data given a tree and a model, using the pruning algorithm. The pruning algorithm is a dynamic programming algorithm that allowed the computation of a likelihood to be completed in polynomial time instead of exponential time.

These models were further refined by Felsenstein (1984) in Phylip, and by Hasegawa et al. (1985). These combined the Kimura and Felsenstein 1981 models to allow both unequal base frequencies and allowed for a bias of transitions to transversions. Finally, there is the general time-reversible(GTR) model which allows for every type of mutation to have a different rate and for different base frequencies, so long as time reversibility is maintained. In other words,

$$\pi_j P(i \to j) = \pi_i P(j \to i), \tag{2.10}$$

where π_j is the stationary frequency of base j and $P(i \to j)$ is the probability that a mutation from i to j happened. Swofford and Olsen (1990) and later Felsenstein (2004) wrote comprehensive reviews of these models. Although it may seem like the most complicated GTR model is the best model to use, often this model is over parameterized.

Each of these nucleotide evolution models can be thought of as a continuous time Markov process with an instantaneous rate change matrix Q. The off diagonal columns of this matrix have the relative frequency of a mutation from nucleotide i to nucleotide j. Where i is the row of the element and j is the column. Since all these models are time reversible, Q is symmetric (i.e. $Q_{i,j} = Q_{j,i}$).

The probability of change after some time t can be expressed as a matrix exponential

$$P(i|j,t) = P(j|t_0)e^{Qt} \tag{2.11}$$

Given a tree, one can calculate the probability of the nucleotide sequences at the tips of the tree using Felsenstein (1981)'s pruning algorithm.

2.6 Markov Chains and Markov Chain Monte Carlo

In order to integrate the Felsenstein Equation and it's Bayesian counterpart, I have used Markov Chain Monte Carlo. In this section, I will demonstrate the properties of Markov Chains.

A Markov chain is a stochastic process with the Markov property. The Markov property states that if at any time the probability of being in a state depends only on the previous state,

$$P(X_n|X_{n-1}, X_{n-2}, X_{n-3}, \ldots) = P(X_n|X_{n-1}). \tag{2.12}$$

Here X_n is the state of the Markov chain and n is an integer time or index. An example of a Markov chain is shown in figure 2.6. This could be single site on a genome evolving through time. The time step can be considered to be one generation. At each time step a nucleotide can either stay the same or there is a chance that it can mutate to another nucleotide.

It is often the case that the state space(i.e. possible values for X) is finite. In this case, the process can be described by a transition matrix, Q.

$$Q_{i,j} = P(X_{n+1} = j|X_n = i), \tag{2.13}$$

for all n.

This allows for easy calculations. For example, if Q represents a 4-state Markov chain, and at time $n = 0$ the probability of being in each state is represented by the vector P_0 then the probability of being in each state 5 time steps later is P_0Q^5.

Time-Continuous Markov process. A time-continuous Markov process is a Markov Chain that is defined on a continuous time scale. This means that in equation 2.12, n is free to take non integer values. It is still the case that probabilities of a chain being in any state are dependent only on the most recent previous state recorded and not any states before that.

One can think of a time-continuous Markov process as a limit of a discrete-time Markov chain. The Q matrix for a discrete time process can be rewritten as $Q = I + Q'$, or $Q' = Q - I$. Here, I is the identity matrix. Since all the rows of the original Q matrix summed to one, this time they sum to zero and the diagonal elements are

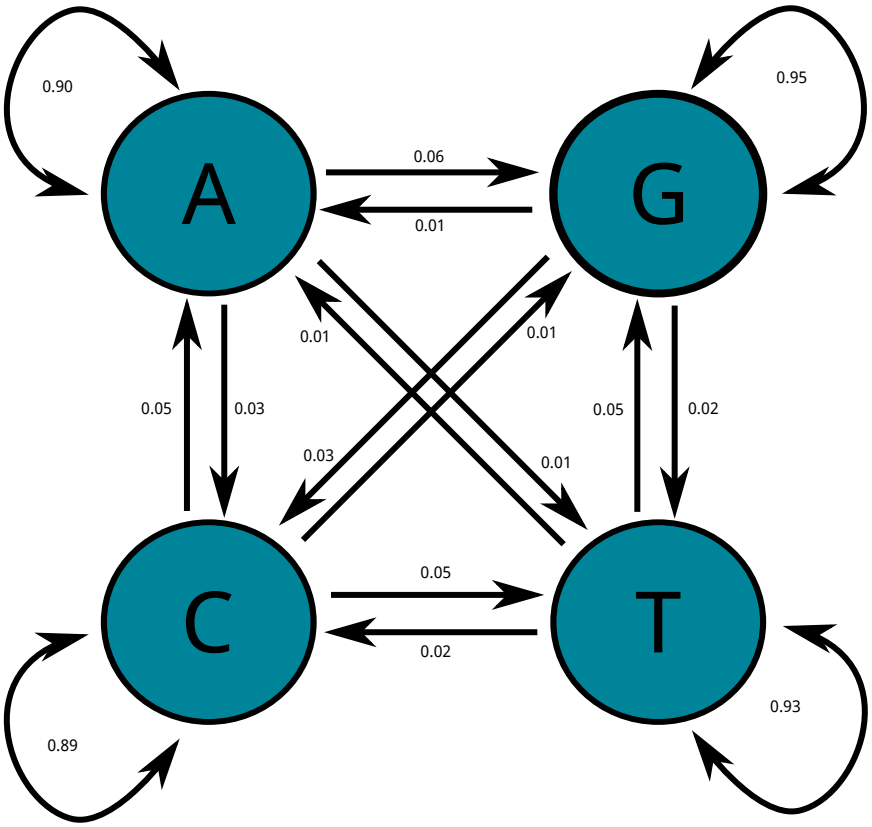


Figure 2.6: An example of a Markov Model. The probability of following each arrow at each time step is shown.

the negative sum of all the other elements in the same row. If one decided that now we want this discrete process to have similar properties to the original process, but instead have increments of time happen twice as often(i.e. instead of 1,2,3,4 do 1,1.5,2.0,2.5,3.0), one could create a new Q matrix in the following way, $Q = I + Q'/2$. However even though this process is discretized more finely, we may still be interested in the same time period of 1. In this case the matrix of interest is $(I + Q'/2)^2$. In order to arrive at a continuous-time Markov process, a limit is taken to discretize more and more finely the discrete time chain.

$$\lim_{n\to\infty} \left(I + \frac{Q'}{n}\right)^n = \sum_{k=0}^{\infty} \frac{1}{k!} Q'^k = e^{Q'} \tag{2.14}$$

The infinite sum is just an expansion of the multinomial product under the limit, while the matrix exponential is defined by that sum. The infinite sum is rarely used to calculate these matrix exponentials because the convergence rate is very slow. There are many different ways to exponentiate a matrix depending on the problem being solved and the nature of the matrix Moler and Loan (2003) have a great review of most of them.

Often, when calculating probabilities using a continuous-time Markov process, one has arbitrary length time intervals. Equation 2.14 changes to $e^{Q't}$ where t is a scalar that represents the amount of time that has passed.

Stationary Distributions. Stationary distributions are important attributes of Markov processes. During the simulation of a Markov process, the state of this process will change many times. After a long enough time period some pattern will emerge. There will be states that are visited more often and states that are seldom visited. In the limit, it is how often these states are visited that forms the stationary distribution. Mathematically, π is a stationary distribution vector if and only if

$$\pi_j = \sum_i \pi_i Q_{ij} \tag{2.15}$$

Q is the same as before for the discrete case or it can be $e^{Q't}$ in the continuous time version. This is true for any t. For nucleotide models of evolution the π vector is often referred to as base frequencies. These base frequencies are often estimated empirically from the data.

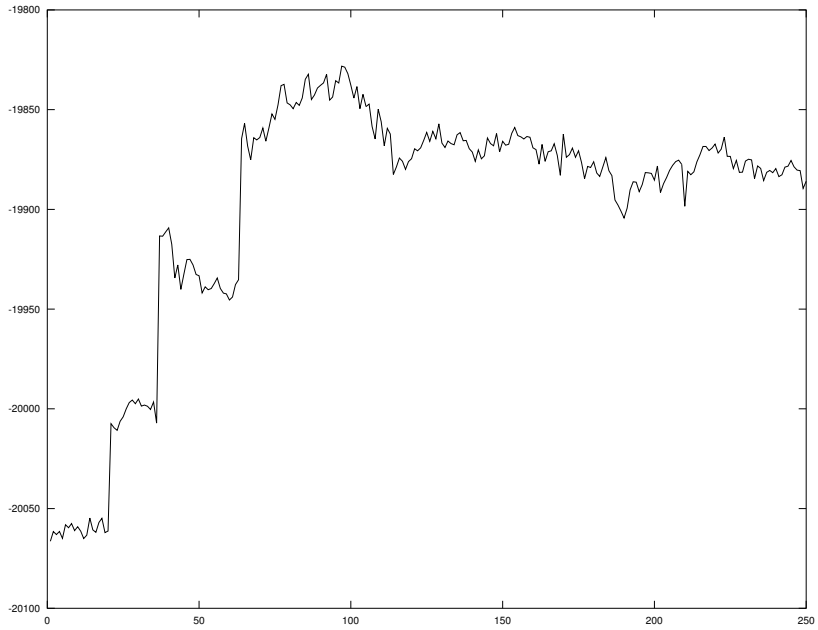


Figure 2.7: Trace plot of the begining of an MCMC run. The y-axis shows the value of the scaled posterior and the x-axis shows the number of iterations of this MCMC algorithm

2.6.1 Markov Chain Monte Carlo (MCMC)

The MCMC method is a sampling scheme designed to allow the sampling of arbitrary distributions. MCMC relies on being able to construct a Markov Process with a stationary distribution that is the same as the distribution we wish to sample from. Most MCMC schemes require what is known as a burn-in. MCMC schemes require a sample form the function of interest in order to sample again from the function. This circular problem is resolved by starting with a value that is not a sample from the correct distribution (an arbitrary value or an educated guess). Then by running MCMC algorithm long enough it will eventually sample from the correct distribution. This initial sampling period is known as the "burn-in". It can be difficult to predict exactly how long to burn-in. Firgure 2.7 shows an example of a burn-in.

Typically, when using MCMC to sample from a function, we are interested in estimating that density function. There are many different ways to estimate such a density but the most common and simplest is a histogram. When using a histogram, typically the area from the largest sample to the smallest sample is split up into equal sized sets called bins. Whenever a sample falls within a bin, the size of that bin is

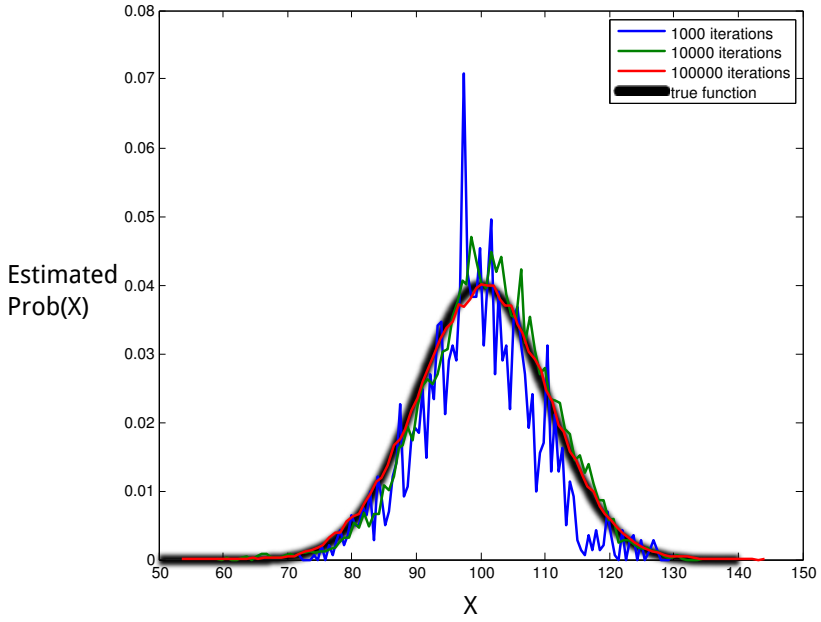


Figure 2.8: Plot showing the convergence of an MCMC algorithm run for a different ammount of iterations.

increased by one. A histogram is a bar graph showing the size of these bins. In Figure 2.8, I demonstrate the convergence of an MCMC algorithm. In this case I know the true function, and it is obvious how the estimate converges to the true function. Due to the lack of any good convergence diagnostics, often the "eyeball" test is employed. If a function looks smooth and multiple runs look similar then it is usually safe to assume convergence.

Metropolis-Hastings. The Metropolis-Hastings algorithm first proposed by (Metropolis et al., 1953) and refined for a more general case by (Hastings, 1970) allows sampling from a function $P(X)$ which only need to be known up to a constant. This algorithm also requires sampling from function $G(X'|X)$, this function updates the current sample to a new one. This is typically done using some sort of perturbation.

The way this algorithm works follows. Given a sample from $X \sim P(X)$, a new sample is chosen from a perturbation distribution(G) $X' \sim G(X'|X)$. This sample is either accepted or rejected. The probability of acceptance is

$$\min\left\{\frac{P(X')G(X|X')}{P(X)G(X'|X)}, 1\right\} \tag{2.16}$$

If the sample is rejected, the old value of X is retained and that becomes the new sample. If the sample is accepted then X' becomes the new sample. Even if $G(X|X') = G(X)$ (i.e., a new proposed X does not depend on the previous one) this algorithm always give correlated samples. This is because there is always a chance that the new proposed value will be rejected.

The sliding window proposal is a very common perturbation function(i.e., G). The way this function works is by sampling a random uniform interval around the current X as such

$$u \sim U(0, 1) \tag{2.17}$$

$$X' = (u - 0.5) \cdot w \tag{2.18}$$

w is the length of the uniform random interval. Figure 2.9 shows pictorially how a Metropolis Hastings with a sliding window proposal could work.

The width of the random uniform interval is considered a tuning parameter, it will make a difference in how quickly the algorithm converges and an appropriate value should be determined before the MCMC run or during the burn-in. A small window width will lead to the new parameter being accepted much of the time since the new value will be close to the old value. However, a small window width will not allow much exploration of the function since with each update the sample will have similar probability to the old sample. A large window width may lead to the new parameter being rejected much of the time, since new proposals may be far away from the current point which after burn-in is presumed to be in a high probability area.

The quantity $\frac{G(X|X')}{G(X'|X)}$ is also known as the Hastings ratio. If the perturbation function that proposes new values is biased in some way, this ratio in the acceptance probability will correct it. For the sliding window proposal, the Hastings ratio is one. The size of the uniform distribution around the current sample does not change and both samples will either be within each others windows or they will not. A Metropolis Hastings algorithm with a sliding window proposal requires at least one additional calculation of the scaled probability function to produce a new value. It is possible that it will never produce a new value, but this is rare and a sign that the width of the sliding window is too great. Assuming a 50% acceptance rate, this algorithm requires a new evaluation of the probability function on average twice for each sample.

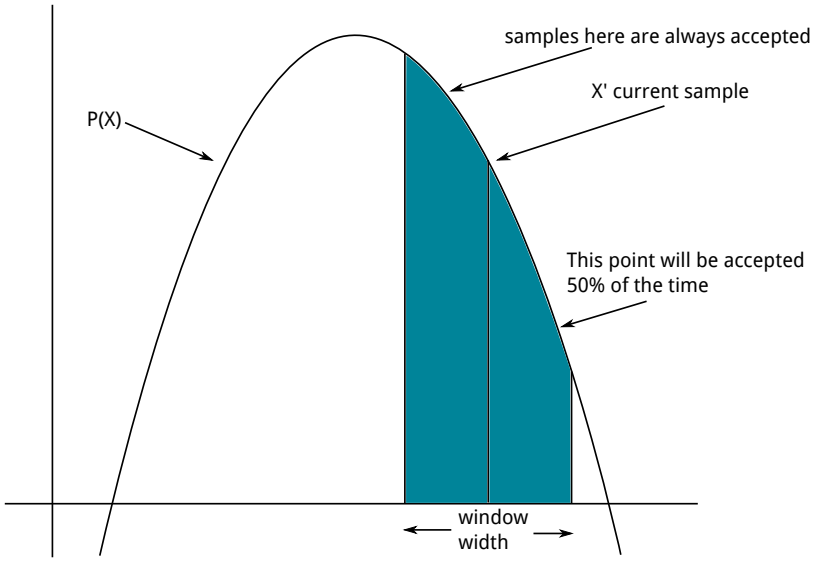


Figure 2.9: Metropolis Hastings with the sliding window update, the colored area shows where candidate new samples will be drawn from.

Slice Sampling. Slice sampling is an MCMC algorithm proposed by Neal (2003). This algorithm has some advantages over Metropolis-Hastings in that it never rejects and is always able to create a new sample. Again sampling is from $P(X)$ and it only needs to be calculated up to a constant. First a random variable y is chosen uniformly from $(0, P(X))$, where X is the previous sample. Next, a new X is chosen uniformly from the region where $P(X) > y$. It is not always obvious how to compute and/or sample from such a region. Instead boundaries are computed. First, a stick length is arbitrarily chosen (The meaning of the stick length will become clear.) Then $P(X+l)$ is computed. If this quantity is less than y, then $X+l$ is the right boundary. Otherwise, $f(X+2l)$ is computed and checked if it is a suitable right boundary. This continues until a suitable right boundary is found. This process is repeated except this time subtracting from the original sample to find a suitable left boundary. The new sample is picked from a uniform distribution formed by the left and right boundary points. If the new point is greater than or equal to y, it is accepted as a new sample, otherwise depending on which side this new point lies, the boundaries are made smaller. Figure 2.10 shows a visual demonstration of slice sampling.

Slice sampling has some advantages and disadvantages over Metropolis-Hastings. A suitable perturbation function does not need to be found. Thus, regardless of the shape of the function, slice sampling will get reasonable samples, though it may take

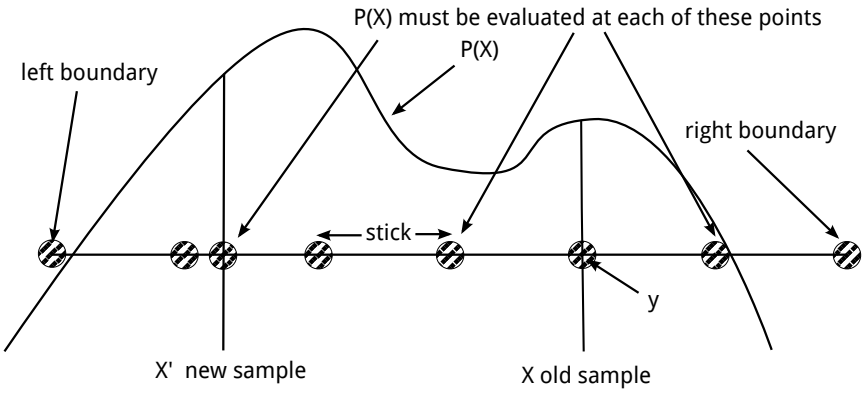


Figure 2.10: A pictorial view of slice sampling

time to get them. Adjusting the length of the stick during an MCMC run is valid, whereas adjusting the window size on a sliding window proposal is not(though it can be done during burn-in). Due to the nature of slice sampling, a way to slice-sample trees has not been published.

2.6.2 Bayesian Inference

The goal of Bayesian inference is to estimate a posterior function. This is in contrast to Likelihood based inference where the goal is a point estimate of a parameter of interest called the Maximum Likelihood Estimate(MLE for short). Often, a confidence interval is estimated along with it. Calculating these confidence intervals can be very time consuming in multiple dimensions and often interactions between parameters are overlooked, and instead, one dimensional confidence intervals called profiles are created. This is still in contrast to frequentist statistics which are typically used to test one hypothesis against another and estimate a p-value corresponding to how repeatable an experiment is.

The posterior function takes the following form.

$$P(\theta|D) = \frac{P(\theta)P(D|\theta)}{P(D)} \tag{2.19}$$

Here θ is the parameter or parameters of interest and D is the data. The probability $P(D|\theta)$ is the probability of simulating the data from the parameters. $P(D)$ is a normalizing constant. It represents the probability of simulating the data under the model. I will elaborate on this term in the next section(2.7). $P(\theta)$ represents the

prior belief about the parameters before running the experiment. Felsenstein (2004) showed that, for certain problems it can be difficult to choose a prior that does not bias the results of the experiment. Frequently, a uniform distribution on a range is used as a prior. In this case the shape of the posterior on that range should be the same as the likelihood function.

In phylogenetic and population genetic literature, MCMC is a method often employed in estimating the posterior density function. Samples are taken from the function using MCMC and a histogram is made to show an estimate of the parameters.

2.7 Bayes Factors

Bayesian inference lends itself very well to model selection. Equation 2.19 omitted the dependence on the model. In fact, all the terms in this equation are conditional on the model being used for the inference.

$$P(\theta|D, M) = \frac{P(\theta|M)P(D|\theta, M)}{P(D|M)} \tag{2.20}$$

Here, M represents the model being used for estimating the posterior. While θ represents all the parameters in the model. Of particular interest is the term $P(D|M)$. This is called the marginal likelihood of the model. Since this is a likelihood function, once again it can be reversed to make a posterior.

$$P(M|D) = \frac{P(M)P(D|M)}{P(D)} \tag{2.21}$$

This requires a prior for different models. Unless there is a reason to prefer one model over another, typically all models have a uniform prior.

Of particular interest is the comparison between two models. Typically the following ratio is computed

$$B_{12} = \frac{P(D|M_1)}{P(D|M_2)} \tag{2.22}$$

Here B_{12} is called the Bayes Factor. It is a summary of the evidence for one model (M_1) versus the other (M_2). Table 2.2 shows an interpretation provided by Jeffreys (1961) for different values of B_{12}.

Table 2.2: A suggested interpretation of the strength of Bayes Factors of one model versus the other. Jeffreys (1961).

$\log_{10}\lvert B\rvert$	$\lvert B\rvert$	Evidence
0 to $\frac{1}{2}$	1 to 3.2	Not worth more than a bare mention
$\frac{1}{2}$	3.2 to 10	Substantial
1 to 2	10 to 100	Strong
>2	>100	Decisive

Efficient computation of Bayes factors is difficult. The most obvious way of computing a Bayes factor is Monte Carlo estimation. Here I write a Monte Carlo estimator(nb. I omit the dependence on the model for clarity).

$$P(D) = \int P(\theta)P(D|\theta)d\theta \approx \frac{1}{k}\sum_{i=1}^{k} P(D|\theta_i) \ , \ \theta_i \sim P(\theta) \tag{2.23}$$

This samples from the prior and takes an arithmetic average of the likelihoods. Unfortunately this converges slowly. Most of the values of θ are poor fits to the likelihood function. Since in a typical Bayesian analysis, samples are drawn from a posterior. Newton and Raftery (1994) suggest the following harmonic mean estimator which makes use of these samples.

$$\theta_i \sim P(\theta|D) \ , \ \frac{1}{\frac{1}{k}\sum_{i=1}^{k}\frac{1}{P(D|\theta_i)}}. \tag{2.24}$$

Working backwards and using Bayes' theorem one can indeed verify what the sum is estimating.

$$\frac{1}{k}\sum_{i=1}^{k}\frac{1}{P(D|\theta_i)} \approx \int_\theta P(\theta|D)\frac{1}{P(D|\theta)}d\theta = \int_\theta \frac{P(\theta)P(D|\theta)}{P(D)}\frac{1}{P(D|\theta)}d\theta \tag{2.25}$$

$$= \frac{1}{P(D)}\int P(\theta)d\theta = \frac{1}{P(D)} \tag{2.26}$$

This is a very convenient estimator since posteriors are typically estimated by sampling. Unfortunately, this estimator can give poor estimates (Neal, 2008), since the variance in this estimate can be infinite. For population genetics problems involving the coalescent, Beerli and Palczewski (2010) showed that this is an inadequate estimator.

Lartillot and Philippe (2006) proposed thermodynamic integration as a method to estimate the marginal likelihood. Beerli and Palczewski (2010) used precisely this method to get good estimates of marginal likelihoods. This method draws a continuous path from no model(not to be confused with null model), to the model of interest.

$$q_\beta(\theta) = p(D|\theta)^\beta p(\theta) \tag{2.27}$$

Here q_β is an unnormalized density function. q_0 would then be the prior, while q_1 would be the unnormalized density of the posterior of interest. In order to compute the marginal likelihood the following integration is performed numerically.

$$\int_0^1 E_\beta[\ln p(D|\theta)]d\beta \tag{2.28}$$

To estimate this integral, we use MCMC to sample from various p_β. Migrate already used a variant of MCMC called Metropolis Coupled Markov Chain Monte Carlo(Geyer, 1991). This variant of MCMC concurrently samples from different distributions. In our case, we were already sampling from the correct distribution, p_β. In order to estimate this integral, we developed a quadrature specific to our problem.

CHAPTER 3

POPULATION MODEL COMPARISON USING MULTI-LOCUS DATASETS

3.1 Preface

The following is from a book chapter that is in pre-publication. This work is based on the model inference work I did with Peter Beerli. I was responsible for the mathematics, formulas, algorithms, and the majority of the writing.

3.2 Introduction

Bayesian inference has changed the study of phylogenetics and population genetics. Just a few years ago researchers using probabilistic methods had to justify using such methods rather than parsimony-based tree inferences in phylogenetics and allele-frequency based methods in population genetics. Molecular phylogenetics seems to be more progressive than population genetics in accepting Bayesian or maximum-likelihood methods because today it is common to find phylogenetic reports that only employ probabilistic methods, in contrast, population genetics reports that do not report summary statistics along-side probabilistic methods are rare. We assume this is mostly based on the fact that in phylogenetics usually only one marker, a long stretch of DNA, was collected from many different species; this made it rather simple to develop statistical methods and focus on the mutation model that changes the sequence data over evolutionary time, leading in turn to development of a large number of different mutation models and variants. These models considered, for example, site rate variation and coding versus non-coding sequences. Population genetics, on the other hand, focused on allele frequencies among many sampling

locations of a single species. Once sequencing was feasible for many individuals, however, it became obvious that sequencing the same stretch of DNA from many individuals in a single population contributes little additional information because most individuals are identical by descent. The allozyme era of the '80s revealed, however, that populations show many differences if we are willing to look at many loci. This led to a search for cheap markers, such as microsatellites and single nucleotide polymorphisms (SNPs). Recently, studies on non-model organisms that use many stretches of DNA sequence have emerged.

Approaches in biogeography have blurred the boundaries between phylogenetics and population genetics, from within-species sampling for strict phylogenetics purposes to the effects of variability within and between species samples. In population genetics, methods for explicitly modeling population divergence and combinations of other population genetic forces, such as changes of population size through time and migration patterns among populations, are beginning to emerge.

For many of these analyses, the use of more data improves the accuracy of the inference. One can increase the number of individuals, the number of populations or species, the lengths of the sampled sequences, and the number of unlinked loci. Felsenstein (2005), Pluzhnikov and Donnelly (1996), and Carling and Brumfield (2007) have shown that in a population-genetic framework the information gained from increasing the number of individuals is limited: a sample of many individuals from the same population will reveal many close relatives – the samples are not independent from each other. Increasing the number of populations and species often does not help because we also increase the size of the model that our inference needs to solve. Increasing the sequence lengths may help, but eventually the assumption that the sites in the sequence all have the same evolutionary history is violated because of recombination. We would need to treat the left and the right ends of a large stretch of a sequence as different, unlinked loci. This leaves the last option, increasing the number of unlinked loci, as a natural way to improve the analysis. In phylogenetics many sequences are now partitioned to allow different mutation models, but few methods allow independent analysis of each partition, for example, by running independent Markov chains for each partition to sample independent groups of trees.

In this chapter, we will focus on population genetics analyses, but believe that the same problems and solutions will hold for phylogenetic inferences with multilocus datasets.

In current applications in population genetics, the number of independent loci that can be used has increased considerably, many of the new genome-scale approaches use single nucleotide polymorphisms and summary statistics either based on allele frequencies or other summarizing tools such as principal components or similar approaches. Coalescence-based Bayesian approaches have not yet been tested with thousands of loci, but in principle, programs like MIGRATE (Beerli, 2006), or LAMARC (Kuhner, 2006) can be run with an unlimited number of loci. Likelihood inference for a multilocus dataset can be run in parallel because, assuming the loci are independent, the calculation for each locus can be easily parallelized, and the final result is a simple combination of the individual results. We have run MIGRATE successfully using 10,000 loci using a two-population model on a computer cluster with 256 nodes within 2 hours.

In Bayesian inference, the independent calculations of the posterior distributions for each locus is simple, but, in contrast to the combination of maximum likelihood estimates over loci, the product of these posterior distributions leads to an overuse of the prior. Correction for this overuse allows us to calculate the posterior distributions independently on different computers or CPU cores, therefore improving the speed of analysis.

3.3 Bayesian inference of independent loci

Bayesian inference is the process of making statements of belief about parameters of interest based on a set of data. At the core of Bayesian inference is Bayes formula

$$\mathrm{P}(\theta|D) = \frac{\mathrm{P}(\theta)\mathrm{P}(D|\theta)}{\mathrm{P}(D)} \tag{3.1}$$

The goal of the calculations is to make a statement of the parameter θ in the light of the data D: the posterior distribution of the parameter. We use θ as a placeholder for a single parameter or a vector or parameters. The $\mathrm{P}(\theta)$ is the prior belief about the parameter θ, for example, if we are interested in estimating the probability of tossing head with a particular coin, we could assume that all probabilities with in the range of 0 and 1 are feasible and equally likely, or we could believe that usually a coin is not manipulated and so the estimate should be close to 0.5, we could use a distribution that peaks at 0.5 and has lower probability at 0 and 1, for example a

beta distribution with parameters $\alpha = \beta = 2$. The denominator is the probability of the data, which is equivalent to the integral of the numerator over the range of θ. For most inference purposes this is simply a scaler of the posterior probability.

In a typical Bayesian analysis, the posterior distribution is presented as the result. Often we cannot calculate the posterior analytically, and we resort to stochastic methods, for example Monte Carlo. For tree-based inferences in phylogenetics and population genetics we sample parameter values from this posterior using Markov Chain Monte Carlo (MCMC). These samples are collated and a histogram representing the final posterior is created. The popularity of MCMC stems from the fact that to find the relative posterior distribution we can ignore the denominator in formula 3.1 because MCMC uses the ratio of previous and current state in the Markov chain (Metropolis et al., 1953).

Unfortunately for model comparisons we need to calculate the value of the denominator which is commonly referred to as the "the probability of the data." However, placing a probability value on the data can be counterintuitive, therefore it is sometimes called "the probability of simulating the data." To gain further insight we rephrase formula 3.1 as

$$\mathrm{P}(\theta|D) = \frac{\mathrm{P}(\theta)\mathrm{P}(D|\theta)}{\int_\theta \mathrm{P}(\theta)\mathrm{P}(D|\theta)d\theta}. \tag{3.2}$$

or we can be even more explicit by adding the inference model M

$$\mathrm{P}(\theta|D, M) = \frac{\mathrm{P}(\theta|M)\mathrm{P}(D|\theta, M)}{\mathrm{P}(D|M)}. \tag{3.3}$$

The probability of the data given the model which is equivalent to the expectation of the likelihood under the prior:

$$\mathrm{P}(D|M) = \int_\theta \mathrm{P}(\theta|M)\mathrm{P}(D|\theta, M)d\theta = \mathrm{E}[\mathrm{P}(D|\theta, M)]_{\mathrm{P}(\theta|M)}. \tag{3.4}$$

Incorporating the model M explicitly into the marginal likelihood emphasizes the implicit assumption of a particular model under which all terms are calculated. We estimate the probability of the data given a model. If one can calculate this marginal likelihood under model A and then calculate it again under model B, we can directly compare model A and model B. The model with the higher marginal likelihood is

Table 3.1: Bayes Factors shown as ratios using the cutoff values devised by Jeffreys (1961) for two models M_1 and M_2.

BF	ln BF	Strength of evidence in favor of M_1
$<$ 1:1	< 0	Negative (Supports M_2)
1:1 to 3:1	0 to 1.1	Barely worth mentioning
3:1 to 10:1	1.1 to 2.3	Substantial
10:1 to 30:1	2.3 to 3.4	Strong
30:1 to 100:1	3.4 to 4.6	Very Strong
$>$ 100:1	> 4.6	Decisive

preferred because it explains the data better. Unlike other methods of model inference such as the likelihood ratio test this method can give a higher likelihood and show preference for a model with less parameters than a more complicated model.

Jeffreys (1961) quantified the degree of support for one model over another by calculating the Bayes Factor

$$\mathrm{BF} = \frac{\mathrm{P}(D|M_1)}{\mathrm{P}(D|M_2)} \tag{3.5}$$

He also indicated an interpretation of the values (Table 3.1). These values assume that the prior on selecting models is uniform, they all have equal prior probability.

The previous exposition assumes that the data D are a single piece of information, but often the data set is structured in one way or another. For example, it is common in phylogenetics to partition the data set into different segments and use different mutation models for each partition; in population genetics it is common to assume that different loci are independent of each other: every locus is an independent replicate of the evolutionary process (model) that lead to the data observed, thus every locus can be thought of as an independent dataset. Since estimates from each locus can be independently obtained it makes sense to parallelize the analysis for high performance computing and run each locus on a different processor in parallel. Beerli (2004) used such an approach for calculating maximum likelihood estimates, but running large numbers of loci independently and then finding the maximum likelihood using

$$\mathrm{P}(D_1, D_2, \ldots, D_n|\theta) = \prod_i^n \mathrm{P}(D_i|\theta) \tag{3.6}$$

simply translating this procedure to Bayesian inference would lead to

$$P(\theta|D_1, D_2, \ldots, D_n) = \frac{\prod_{i=1}^{n} P(\theta|D_i)}{P(D_1, D_2, ..., D_n)} \tag{3.7}$$

Unfortunately, this is incorrect.

Although each of our estimates of the parameter is independent, they are thought to be estimating the same parameter value. Overuse of the prior is a concern. Instead evaluate correctly the following.

***Theorem* 1.** *The posterior*

$$P(\theta|D_1, D_2, ..., D_n) = \frac{P(\theta)\prod_{i}^{n} P(D_i|\theta)}{\int_\theta P(\theta)\prod_{i}^{n} P(D_i|\theta)d\theta} \tag{3.8}$$

with independent locus data $D_1, D_2, .., , D_n$*, and a set of parameters* θ *can be calculated by*

$$P(\theta|D_1, D_2, ..., D_n) = \frac{P(\theta)^{1-n}\prod_{i}^{n} P(\theta|D_i)}{\int_\theta P(\theta)^{1-n}\prod_{i}^{n} P(\theta|D_i)d\theta} \tag{3.9}$$

Proof. Expanding $P(\theta|D_i)$ in (3.9) leads to

$$P(\theta|D_1, D_2, ..., D_n) = \frac{P(\theta)^{1-n}\prod_{i}^{n} \frac{P(\theta)P(D_i|\theta)}{\int_\phi P(\phi)P(D_i|\phi)d\phi}}{\int_\theta P(\theta)^{1-n}\prod_{i}^{n} \frac{P(\theta)P(D_i|\theta)}{\int_\phi P(\phi)P(D_i|\phi)d\phi}d\theta}. \tag{3.10}$$

The integrals over ϕ cancel, so that

$$P(\theta|D_1, D_2, ..., D_n) = \frac{P(\theta)^{1-n}\prod_{i}^{n} P(\theta)P(D_i|\theta)}{\int_\theta P(\theta)^{1-n}\prod_{i}^{n} P(\theta)P(D_i|\theta)d\theta}. \tag{3.11}$$

Moving the $P(\theta)$ in (3.11) out of the products results in equivalence of (3.8) and (3.9). □

Bayes factors offers a convenient tool for comparing different population models without requiring that models be nested. In usual MCMC-based Bayesian inference the marginal likelihoods are not computed because these normalizing weights cancel in comparisons during the run. They need to be computed and recorded, however, when the combined marginal likelihoods need to be reported. We must evaluate the

denominator of (3.8)

$$P(D_1, D_2, ..., D_n|M_i) = \int_\theta P(\theta|M_i) \prod_i^n P(D_i|\theta, M_i) d\theta. \tag{3.12}$$

It would be tempting to calculate the marginal likelihoods for each D_i independently, but this is incorrect. Even though each dataset is an independent sample, they are thought to come from the same set of parameters.

$$P(D_1, D_2, ..., D_n|M_i) \neq P(D_1|M_i)P(D_2|M_i)...P(D_n|M_i) \tag{3.13}$$

The interdependence of the loci based on the same set of parameters must be taken into account.

Theorem 2. *The combined marginal likelihoods over all independent data blocks can be calculated as a product of independently calculated marginal likelihoods for each data block and a a term that depends on the model and the data.*

Proof. The combined estimator of the posterior distribution is

$$P(\theta|D_1, ..., D_n, M_1) = \frac{P(\theta|M_1) \prod_i^n P(D_i|\theta, M_1)}{P(D_1, ..., D_n|M_1)}. \tag{3.14}$$

Converting the likelihoods using posteriors on the right:

$$\begin{aligned} P(\theta|D_1, ..., D_n, M_1) &= \frac{P(\theta|M_1) \prod_i^n P(\theta|D_i, M_1) P(D_i|M_1)}{P(\theta|M_1)^n P(D_1, ..., D_n|M_1)} \\ &= \frac{\prod_i^n P(\theta|D_i, M_1) P(D_i|M_1)}{P(\theta|M_1)^{n-1} P(D_1, ..., D_n|M_1)}, \end{aligned} \tag{3.15}$$

moving $P(D_1, ..., D_n|M_1)$ to the left and $P(\theta|D_1, ..., D_n, M_1)$ to the right results in

$$P(D_1, ..., D_n|M_1) = \prod_i^n P(D_i|M_1) \frac{\prod_i^n P(\theta|D_i, M_1)}{P(\theta|M_1)^{n-1} P(\theta|D_1, ..., D_n, M_1)}. \tag{3.16}$$

The fraction has to be a constant with respect to θ because both the product of the individual marginal likelihoods and the combined marginal likelihood on the left are constants with respect to θ:

$$K = \frac{\prod_i^n P(\theta|D_i, M_1)}{P(\theta|M_1)^{n-1} P(\theta|D_1, ..., D_n, M_1)} \tag{3.17}$$

Moving the combined posterior and integrating both sides with θ leads to a re-expression of K:

$$P(\theta|D_1, ..., D_n, M_1)K = P(\theta|M_1)^{1-n} \prod_i^n P(\theta|D_i, M_1) \tag{3.18}$$

$$\int_\theta P(\theta|D_1, ..., D_n, M_1)K d\theta = \int_\theta P(\theta|M_1)^{1-n} \prod_i^n P(\theta|D_i, M_1) d\theta \tag{3.19}$$

and because

$$\int_\theta P(\theta|D_1, ..., D_n, M_1) d\theta = 1 \tag{3.20}$$

we can evaluate the scaling factor

$$K = \int_\theta P(\theta|M_1)^{1-n} \prod_i^n P(\theta|D_i, M_1) d\theta. \tag{3.21}$$

This allows the calculation of the combined marginal likelihood using independent inferences

$$P(D_1, ..., D_n|M_1) = K \prod_i^n P(D_i|M_1) \tag{3.22}$$

□

3.3.1 What K represents qualitatively

K represents the "agreement" that multiple loci have about the parameters. For example, one locus favors strong migration while another favors weak migration in the same model. We would expect that there is less evidence for that particular model than when the two loci are in better agreement. Figure 3.1 showcases this "agreement" for four different situations with loci. The subfigures show two posterior

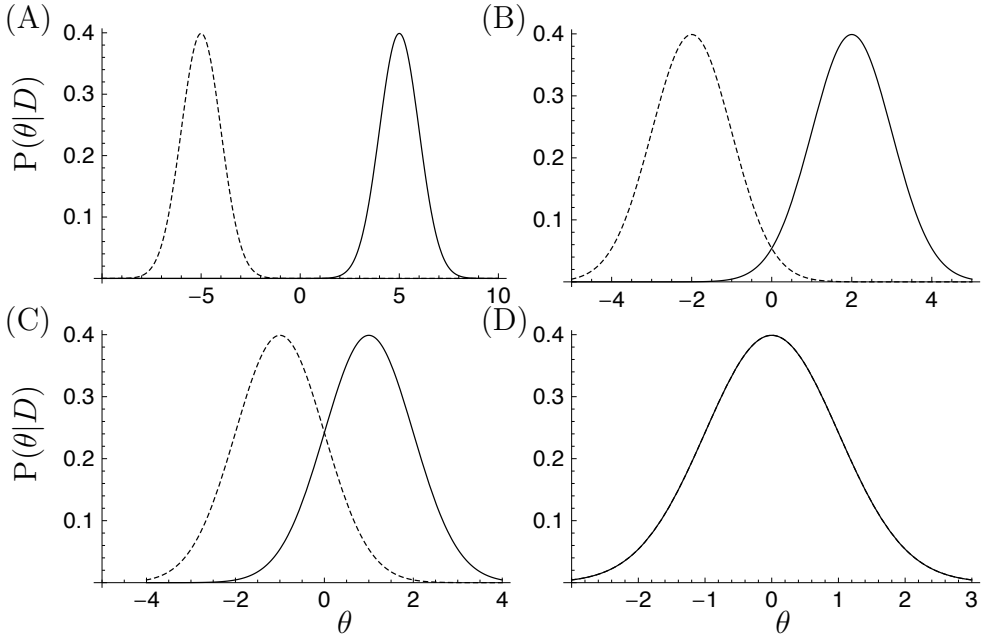


Figure 3.1: Four graphs of possible posterior distributions and their associated K value (3.21) needed to combine the single-locus marginal likelihoods. (A) $\mu_1 = -5, \mu_2 = 5$, $\ln K = -23.2698$; (B) $\mu_1 = -2, \mu_2 = 2$, $\ln K = -2.26978$, (C) $\mu_1 = -1, \mu_2 = 1$, $\ln K = 0.73022$; (D) $\mu_1 = 0, \mu_2 = 0$, $\ln K = 1.73022$

distributions, one for each locus. In each scenario the posterior for each locus is a normal distribution with one parameter μ with variance 1.0.

$$P(\theta|D,\mu) = \frac{e^{-\frac{1}{2}(\theta-\mu)^2}}{\sqrt{2\pi}} \tag{3.23}$$

We will truncate this distribution by assuming that the prior used is a uniform from -10 to 10. The parameter μ can vary among loci. Therefore, the posteriors can range from almost no overlap to being right on top of each other. When the means are very different K is very small, when the means are similar, K becomes larger. This suggest that with many loci, where we may have more potential for disagreement of the posterior distribution estimate among the loci, K will be low. By extension, with many parameters we expect also more disagreement and lower K values.

3.3.2 Calculating K

Although equation 3.21 gives us a formula for K it is not an easy quantity to estimate. If θ were a one-dimensional parameter, the estimation of K would be a trivial one-dimensional integration. However, in population genetics, K can range from a two-dimensional θ in the case of two similarly sized populations exchanging similar numbers of migrants to high-dimensional θ, such as 400 in the case of a 20-population model, where every population can exchange varying amounts of migrants (Beerli and Palczewski, 2010). In a multi-dimensional case, equation 3.21 can be rewritten like this

$$K = \int_{\theta_1} \int_{\theta_2} \dots \int_{\theta_n} \psi^{1-n} \prod_{i}^{n} \mathrm{P}(\theta_1, \theta_2, \dots, \theta_m | D_i, M_1) d\theta_1 d\theta_2 \dots d\theta_n \tag{3.24}$$

where

$$\psi = P(\theta_1, \theta_2, \dots, \theta_m | M_1) \tag{3.25}$$

is the prior distribution. The naive way to estimate this would be to take the estimated samples from an MCMC run, bin them to create a large multi-dimensional histogram and then sum over all the terms.

$$K \approx \sum_{\theta_1} \sum_{\theta_2} \cdots \sum_{\theta_m} \psi^{1-n} \prod_{i}^{n} P(\theta_1, \theta_2, \dots, \theta_m | D_i, M_1)(\Delta\theta_1 \Delta\theta_2 \dots \Delta\theta_m) \tag{3.26}$$

Unfortunately, estimating this term by using a full histogram would be very prohibitive. The default number of histogram bins that MIGRATE uses is 1500. In a 400 parameter case this would mean that $1500^{400} = 2.73 \times 10^{1270}$ terms would be used for this sum. This is far greater than the number of atoms in the observable universe($\sim 10^{80}$). This equation uses $O(nh^m)$ terms. Where n is the number of loci, m is the number of parameters, and h is the number of histogram bins. Although MIGRATE can estimate accurate individual parameter estimates with far fewer samples, it will be unlikely to get multiple observations for each histogram bin during an MCMC run. Therefore, we are forced to use a simplification. Although, we know that

parameters have the potential to be correlated, we make the following adjustment,

$$P(\theta_1, \theta_2, \ldots, \theta_m | D_i, M_1) \approx P(\theta_1 | D_i, M_1) P(\theta_2 | D_i, M_1) \ldots P(\theta_n | D_i, M_1). \tag{3.27}$$

Thus, we assume that all parameter posteriors are independent. We also make an additional adjustment of the prior distribution

$$P(\theta_1, \theta_2, \ldots, \theta_m) \approx P(\theta_1) P(\theta_2) \ldots P(\theta_m) \tag{3.28}$$

Although, it is possible to formulate non-independent priors, we only consider this case.

Our equation 3.24 simplifies to the parameter-unlinked

$$K_u = \prod_j^m \int_{\theta_j} P(\theta_j)^{1-n} \prod_i^n P(\theta_j | D_i, M_1) d\theta_j \tag{3.29}$$

$$\approx \prod_j^m \Delta\theta_j \sum_{\theta_j} \left[P(\theta_j)^{1-n} \prod_i^n P(\theta_j | D_i, M_1) \right] \tag{3.30}$$

This equation is $O(nhm)$. Although there is an accuracy trade off, this equation is possible to calculate. The density for each parameter is calculated individually. Each parameter has a contributing term to the scaling factor. If parameters are highly correlated this could pose a problem. We are using this K_u in our program MIGRATE.

The worst case scenario for our approximation is the situation where all parameters are completely linked, this is very unlikely with real data, in particular with our main interest: the inference of population sizes and migration rates in structured populations. Assuming this worst case scenario for two parameters θ_1 and θ_2, we can express one parameter as a linear combination of the other, for example:

$$\theta_2 = c_1 \theta_1 + c_2 \tag{3.31}$$

so that

$$P(\theta_1, \theta_2 | D) = P(\theta_1 | D) \mathbb{1}_{\{\theta_1, \theta_2 | \theta_2 = c_1\theta_1 + c_2\}}(\theta_1, \theta_2), \tag{3.32}$$

where $\mathbb{1}$ is the indicator function. Then formula 3.24 becomes the parameter-linked

$$K_l = \int_{\theta_1} P(\theta_1)^{1-n} \prod_i^n \mathrm{P}(\theta_1|D_i, M_1) d\theta_1. \tag{3.33}$$

Migrate uses a combination of the above formulae. In the case of asymmetric parameters and population sizes that differ among populations, migrate uses the independent version of the formula. In the case of symmetric migration rates or populations with the same size, the parameters are fully linked and the second formula is appropriate.

This is equivalent to computing K for just one parameter. K_l and K_u are at different ends of the spectrum with regards to the true value of K. The square of the correlation (r^2) between two parameters explains how much of the variance of one variable can be explained by the variance of the other variable. Thus we could construct a weighted average

$$K \approx K_l(r^2) + K_u(1 - r^2), \tag{3.34}$$

For more than two parameters one could imagine analysis of variance based approach, but we have not investigated this option.

3.4 Model comparison using our independent marginal likelihood sampler

Bayes factors make it easy to test nested or non-nested models. Accuracy of the marginal likelihood approximation from inferences using MCMC is a great concern. Estimators of the marginal likelihood based on the harmonic mean (cf Kass and Raftery, 1995) have been proven to be unreliable (Fan et al., 2011; Beerli and Palczewski, 2010; Neal, 2008). Our own thermodynamic integration is less sophisticated than those introduced by Xie et al. (2011) and Fan et al. (2011) and may need more computation but does accurately judge models (Beerli and Palczewski, 2010). Analyzing population structure can involve rather complicated models that may have a wide range of parameters. We can count all 'unidirectional' migration

models ignoring population sizes using

$$\sum_{i=0}^{n(n-1)} \binom{n(n-1)}{i}. \tag{3.35}$$

For 4 populations, this results in 4096 models. The numbers of models increases considerably if we allow asymmetric migration and also take into account that some of the sampling locations could be part of the same panmictic population, for example location 1-3 are one population and location 4 is the second.

With population genetics or phylogenetic data, we rarely know the detailed history and therefore we may never know the truth. We use models to help us to understand the history of our samples and the hidden truth. Current research is commonly testing a priori defined hypotheses. Bayesian model comparison will make it easy to compare these a priori models and thus reduces the number of interesting models. Whether the best model is close to the truth is often easy to answer with "No‘". But this model selection process may allow researchers to formulate new models and test those with even more or better data.

We explore the problem of model selection and comparison with a small example that is sufficiently complex. Beerli (2006) used simulated datasets that where generated using a model of 4 populations that exchange migrants in a round-robin scheme, where population P_1 receives migrants from population P_2, P_2 receives migrants from P_3, P_3 receives migrants from P_4, and finally P_4 receives migrants from P_1, we abbreviate this scenario as $P_1 \rightarrow P_4 \rightarrow P_3 \rightarrow P_2 \rightarrow P_1$. We picked the first 10 single locus datasets and generated 3 new datasets, one with two, five , and ten loci, respectively. For each dataset we evaluate six models (Table 3.2). Model H_0 is the model used

Table 3.2: Different models

Model	Parameter	Explanation
H_0	8	true model: $P_1 \rightarrow P_4 \rightarrow P_3 \rightarrow P_2 \rightarrow P_1$
H_1	9	same as H_0 and addition of $P_4 \rightarrow P_2$
H_2	9	same as H_0 and addition of $P_3 \rightarrow P_1$
H_3	1	One panmictic population
H_4	4	Two populations $(P_1, P_3) \leftrightarrow (P_2, P_4)$
H_5	4	Two populations $(P_1, P_4) \leftrightarrow (P_2, P_3)$
H_6	16	Full model, all migration routes

to simulate the datasets, we call it the true model. Models H_1 and H_2 add an additional migration route, they are similar in structure to each other and also should be similar in results because the round-robin scheme does not have a particular source population. Models H_3, H_4, and H_5 are rather different from the truth. Model H_6 uses 16 parameters and represents the most complex model. With sufficient data this model is capable to represent the parameters correctly (Beerli, 2006). We ran each model twice, for relative short runs (about two hours) on the high performance cluster at Florida State University using 32 nodes. Figure 3.2 summarizes these runs.

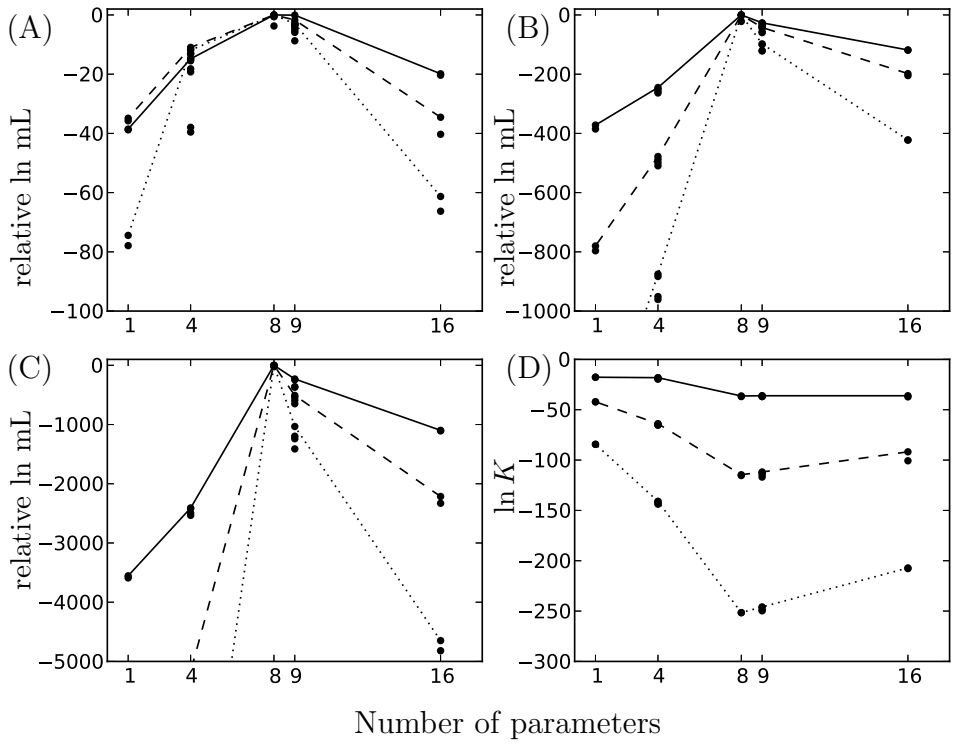


Figure 3.2: Relative log marginal likelihoods (Log Bayes factors) of the models shown in Table 3.2. Each model was run twice. The lines connect the highest scores for each number of parameters and 2 (solid line), 5 (long dashes), and 10 loci (short dashes). (A) dataset had 100 bp per individual; (B) 1,000 bp; (C) 10,000 bp; and (D) Scaling factors for (C).

The difference between all the models is smallest for the two-locus 100 bp dataset and largest for the ten-locus 10,000 bp dataset. With large amounts of data (10,000 bp) the log Bayes factor difference among the models is very large and the true model

H_0 wins even with a small number of loci. Models H_1 and H_2 are most similar to the true model H_0 because they only differ by one additional parameter. This is reflected in the model order from best to worst: $H_0 > H_1 \sim H_2 > H_6 > H_4 \sim H_5 > H_3$. The simulated scenario is clearly structured. This leads to low marginal likelihoods for models that lump populations (models H_3, H_4, and H_5). The full model H_6 has a lower marginal likelihood than model H_0 because 16 varying parameters predict the data poorly. Even with poor data (Figure 3.2A), model H_0 is superior to complex or very simple models, but the marginal likelihood difference from models H_1 or H_2 are small, suggesting that we cannot distinguish H_0 from these models with certainty and we need more data than two loci with only 100 bp.

With ten loci and very short sequences, we may be able to distinguish complex models from each other, but more loci or longer sequences would improve the distinction considerably. For example, using 1000 bp and two loci delivers clearer results than ten loci of 100 bp each.

Our evaluations suggest that even if we omit the true model from the set of tested models, our results still favor models that explain the structured nature of the dataset (models H_1 and H_2).

More data with models that are closer to the true model lead to lower magnitudes of the scaling factors K or $\ln K$ (Figure 3.2D), suggesting that low $\ln K$ indicate a better model fit than high $\ln K$, but we have not explored this relationship in more detail.

3.5 Conclusion

The calculation of Bayes factors for inferences that need MCMC are complex and time consuming. Current approaches for the approximation of the marginal likelihood employ multiple chains with different, static temperatures or a single chain that dynamically changes the temperature to collect likelihood samples. Fan et al. (2011) and Xie et al. (2011) made considerable improvement over our current scheme (Beerli and Palczewski, 2010; Friel and Pettitt, 2008) that allows faster calculation of the marginal likelihood, but only if all the data, for example data partitions, are treated as contiguous blocks with the same genealogy. We provide a framework for combining independently calculated estimates of marginal likelihoods under the assumption that the parameters are independent of each other. This permits analyses

of large-scale biogeographic or population genetic datasets on computer clusters.

CHAPTER 4

CONTINUOUS MIGRATIONS - TPSC

4.1 Motivation

The estimation of migration rates and population sizes are common tasks in population genetics. Both summary statistics and model based approaches have been used to estimate these parameters. Coalesence theory is a robust model based approach that has been shown to be able to estimate population sizes and migration rates. Section 2.4.1 showed a summary of current methods in population genetics.

The structured coalescent is used to estimate migration rates. The complexity of this method grows quickly as the migration rate increases. Figure 4.1 visualizes how quickly the complexity increases as migration rate increases. The tree on the left shows a tree with a low migration rate. There are 4 migration events on the entire tree. One can see that in this example migration does not increase the complexity of tree very much. The tree on the right shows a level of migration that is only one order of magnitude higher. However, the tree is vastly more complex. This increase in complexity only heightens as the migration rate goes up.

The computational cost that each migration event incurs on an inference method is high. Each migration event has a time and a branch on the genealogy associated with it. Current methods attempt to integrate over all possible events by sampling them. That means a program might simulate 20 events on a branch in one step and in the next step re-simulate those 20 events again in a different configuration. This leads to a degradation in program performance as the migration rate increases. In our experiments datasets simulated from a low migration scenario gave good estimates of effective population size and migration rate in a matter of minutes, whereas high migration rate scenarios need hours to deliver estimates.

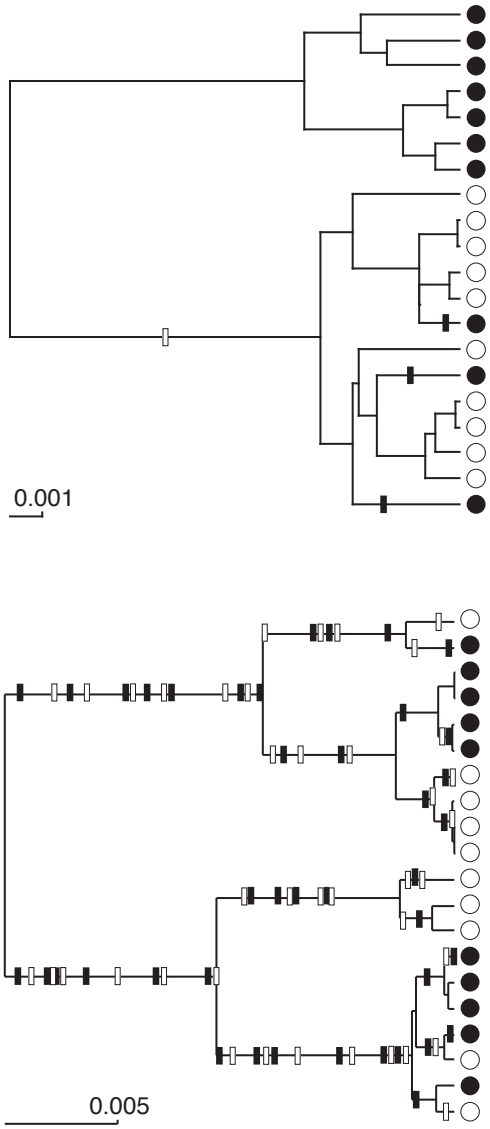


Figure 4.1: Number of migration events in genealogies: (A) genealogy generated with Nm of 0.400 into the population marked with white circles (○) and 0.267 into the population marked with dark disks (●). Migration events on the genealogy are colored according to the receiving population looking forward in time. (B) immigration rates are are 10 times higher.

4.2 Method

I have proposed a continuous model for inferring migration rates. The structured coalescent (Strobeck, 1987) describes genealogies of individuals in multiple populations. It contains discrete events in the form of coalescent and migration events. I describe here a simplification of the structured coalescence by replacing the migration events with transition probability structure. We have dubbed this method TPSC which is short for transition probability structured coalescent.

A single lineage evolving backwards in time can be modeled by a continuous time Markov chain. This has the same result as Strobeck's (Strobeck, 1987) model. Each Markov chain state refers to a population. Thus if the chain is in state 1, that means that the lineage is in population 1. The entries of the Continuous Markov Chain Q matrix are migration rates. For example, Q_{12} would be the rate of migration from population 2 to 1, and represent the probability that an individual in population 1 came from population 2 in the previous generation.

The probability of this lineage being in a population at some time in the past is

$$P(L = i|t) = \sum_j P(L = j|t_0)(e^{Qt})_{ji} \tag{4.1}$$

Here $P(L = i|t)$ is the probability that L a lineage is in population i at time t in the past. $P(L = j|t_0)$ is a similar probability except that it is at the current time. $(e^{Qt})_{ji}$ is the probability of changing from state j to state i in time t, also can be stated as the j'th row and i'th column of the matrix e^{Qt}.

When two lineages are in the same population they will coalesce following an exponential distribution.

$$P(t_{coal}) = \frac{1}{2N} e^{-\frac{1}{2N}t} \tag{4.2}$$

Here t_{coal} represents the time until a coalescence event. This is a homogeneous exponential distribution. A more general case of this distribution is the non-homogeneous exponential distribution.

$$P(t) = \lambda(t) e^{-\int_0^t \lambda(t)dt} \tag{4.3}$$

Equation 4.2 can be put in this form if $\lambda(t) = \frac{1}{2N}$.

Thus, if two lineages are in the same population they coalesce at the rate of $\frac{1}{2N_i}$ for diploids, Where N_i is the effective population size of population i. However if it is unknown which population each lineage is in, this rate will change.

$$\lambda_{i,j,k}(t) = \frac{P(L_i = k, L_j = k|t)}{2N_k} \tag{4.4}$$

Here $\lambda_{i,j,k}(t)$ is the rate of coalescence of lineage i and lineage j in population k. $P(L_i = k, L_j = k|t)$ is the probability that lineage i and lineage j are both in the same state k(which represents a population) at time t. N_k is the size of that population.

It is possible to compute the numerator in equation 4.4, however as Takahata (1988) showed this requires a matrix exponential of a matrix the size of $\binom{n}{K}$, where n is the number of lineages and K is the number of populations. Doing so would defeat the purpose of doing a continuous approximation as the matrix size would grow faster than exponentially and the matrix exponential is an already expensive operation.

Instead I make the following simplification,

$$\lambda_{i,j,k}(t) \approx \frac{P(L_i = k|t)P(L_j = k|t)}{2N_k} \tag{4.5}$$

The probabilities in the numerator can be calculated by equation 4.1. This is a much easier and faster calculation, I will attempt to show under what circumstances this is a good estimate.

4.2.1 Analytic Check

I devised a small scenario to test this approximation. The idea is to create a small scenario that is directly comparable to the TPSC method. There are two lineages in two populations (pop 1 and pop 2). Only migration from 1 to 2 is permitted. I am assuming a haploid species. Thus, looking backwards in time a coalescence is only allowed in population 1. The size of population 1 is N and the size of population 2 is N.

The correct way to model this is to create the following Q matrix and model as a

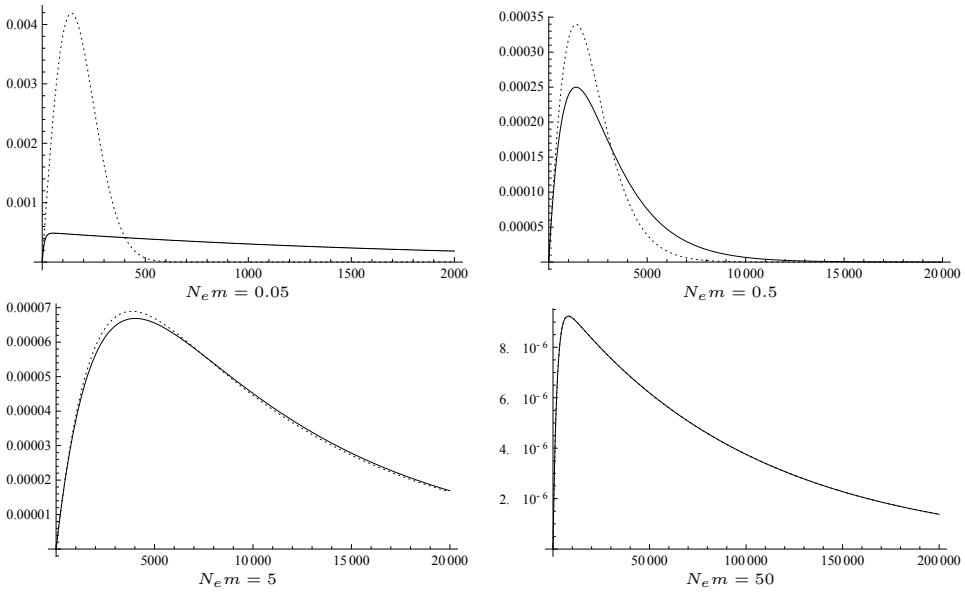


Figure 4.2: Graphs showing the probability density of time to coalescence of two lineages in a two population scenario. The solid line is the exact probability density while the dashed line is my approximation. The approximation works well when migration relative to population size is high.

continuous time Markov Process.

$$Q = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{N} & -\frac{1}{N} & 0 \\ 0 & m & -m \end{bmatrix} \tag{4.6}$$

State 1 represents the coalesced state, it is an absorbing state. State 2 represents both lineages being in pop 1. State 3 is the initial state of two lineages in separate populations. The Probability density of time to coalescence can be computed as

$$P(t) = \frac{d}{dt}\left(\left(e^{Qt}\right)_{(3,1)}\right) = \frac{me^{-mt} - me^{-\frac{t}{N}}}{1 - mN} \tag{4.7}$$

Using my approximate method one first computes $\lambda(t) = \frac{1-e^{-mt}}{N}$ Then the probability density function becomes

$$P(t) = \frac{\left(1 - e^{-mt}\right) e^{-\frac{\frac{e^{-mt}-1}{m}+t}{N}}}{N} \tag{4.8}$$

In figure 4.2, I have compared both functions. In this case scenario it appears like my approximation works very well for high migration and rather poorly for low migration.

One can continue this line of reasoning and compute the total rate of coalescence when there are more than two lineages coalescing.

$$\lambda(t) = \sum_{k=1}^{K} \frac{1}{2} \sum_{i=1}^{n} \sum_{j,j\neq i}^{n} \lambda_{i,j,k}(t) = \sum_{k=1}^{K} \sum_{i=1}^{n} \sum_{j,j\neq i}^{n} \frac{P(L_i = k)P(L_j = k)}{4N_k} \tag{4.9}$$

In both limits of infinite migration(panmixia) and no migration this gives the same results as the standard coalescent. This is the sum of the rate that every lineage coalesces with every other lineage in every population. There is an added factor of $\frac{1}{2}$ due to the double counting of $\lambda_{x,y,z}$ and $\lambda_{y,x,z}$.

Equation 4.9 is computationally expensive and forms a bottle neck in the program because as written this equation is takes $O(n^2K)$ time to compute. Instead I propose the following way.

$$S = \sum_{i=1}^{n} P(L_i = i) \tag{4.10}$$

$$\lambda(t) = \sum_{k=1}^{K} \frac{1}{4N_k} \sum_{i=1}^{n} (S - P(L_i = k))P(L_i = k) \tag{4.11}$$

This is mathematically equivalent however it can be computed in $O(Kn)$ time. This is important because for the program I will describe computing this function proved to be a bottleneck.

If a coalescence takes place, the probability that it is two specific lineages is simply their proportion of contribution to the sum that makes up lambda. This allows the calculation of time to coalescence when there are many lineages.

$$P(\epsilon_{i,j}, t|Q, N) = \lambda_{i,j}(t, Q, N)e^{-\int_0^t \lambda(x,Q,N)dx)} \tag{4.12}$$

I have explicitly added back the dependence on the Q matrix and population sizes inside the λ functions; $\lambda_{i,j}$ is $\sum_{k=1}^{K} \lambda_{i,j,k}$. $P(\epsilon_{i,j}, t|Q, N)$ represents the probability that lineage i and lineage j are the first to coalesce and they coalesced at time t.

I have implemented 4.12 as a function in an inference program. The integral is not solvable analytically. Numerical Integration using Simpson's rule gives a high

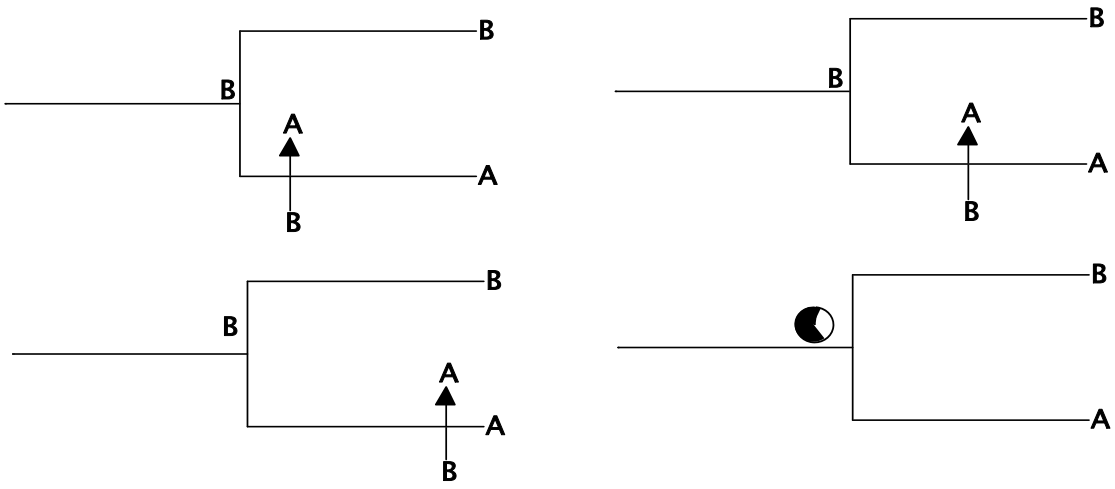


Figure 4.3: The first three trees are ones that a program using the Beerli (1998) method, would sample during the course of the inference. All these trees have one gene in population A and one gene in population B, at the tips. In the first three trees there is a migration from B to A and the coalescence happens in B. The last tree represents my continuous model. Instead of placing individual migrations, only the probability of being in a population is calculated. Trees were drawn by EventTree (Palczewski and Beerli, submitted).

precision with as few as 10 intervals.

A genealogy is consists of $n-1$ coalescence events. To compute the probability of such a genealogy it is only necessary to compute the probability of these events.

$$P(G|Q,N) = \prod_{i=1}^{n-1} P(\epsilon_i, (u_i - u_{i-1})|Q,N) \tag{4.13}$$

This is very similar to the probability of a tree given population parameters in the discrete migration model. Here ϵ_i represents whatever coalescent event happened on G in order from the most recent to the one furthest in the past. u_i represents how much time has elapsed since that event. u_0 is defined to be 0.

4.3 Adaptive Metropolis-Hastings

In order to further improve on the performance of TPSC, I have used methods to allow for tuning my Markov Chain Monte Carlo Chains. According to Roberts and Rosenthal (2009); Gelman et al. (1996) the ideal acceptance rate of an MCMC is somewhere between 20-60%. In order to achieve a reasonable acceptance ratio my algorithms use an adaptive scheme during the burn-in.

Small parameter updates are unlikely to be rejected. This is because most posterior functions are continuous and small parameter changes lead to small changes in the posterior. Large parameter updates are more likely to be rejected for the opposite reason. Large changes to parameters will more likely lead to large changes in the posterior. If the current parameter values are nearly optimal then larger changes will likely get rejected.

In the TPSC program I have used a sliding window proposal to do parameter updates. New parameters are sampled from a normal distribution centered at the old parameter value. It than stands to reason that a rejection might mean that the variance of the normal is too large. While an acceptance might mean that the variance is too small. For this reason during burn-in whenever an acceptance occurs I multiply the variance by a value B which is slightly bigger than 1.0.

$$\sigma_{t+1}^2 = B\sigma_t^2 \tag{4.14}$$

On the other hand when there is a rejection I multiply the variance by b, a value slightly smaller than 1.

$$\sigma_{t+1}^2 = b\sigma_t^2 \tag{4.15}$$

What is left is choosing an appropriate value for B and b. If σ^2 has converged to some value then that means that B and b multiply σ^2 enough times that they cancel each other out. Since they multiply σ^2 in proportion to the number of acceptances and rejections, we can form the following relation between the two.

$$B^{1-R} = b^R \tag{4.16}$$

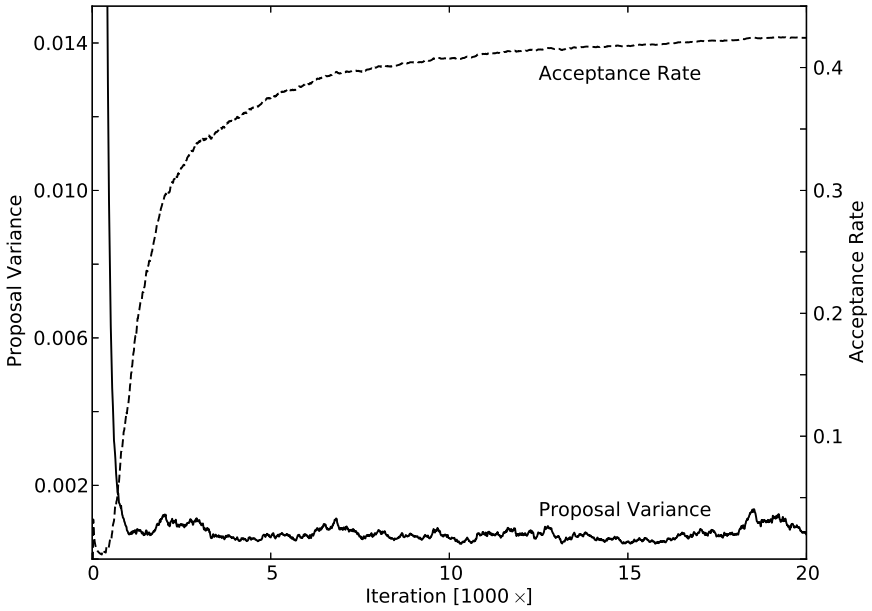


Figure 4.4: An example of the proposal variance adapting to an ideal. The acceptance rate is cumulative and has an asymptote at 0.44.

In TPSC I have chosen $R = 0.44$ as an ideal, like one proposed in Roberts and Rosenthal (2009). I have used an arbitrary value of $b = 0.99$ and solved for B. This insures that the variance is at most 1% away from ideal. Values of b that are closer to 1 would allow for finer convergence, while values of b further from 1 would allow for faster convergence. I have used different values of σ^2 for each parameter. That way the sampling method is tuned on a per parameter basis. Thus each parameter converges to the same acceptance ratio. It is important to note, I have not proved that this will converge, nor that if it does converge that it will do so within the burn in. However, the worst case scenario for this method is not an incorrect solution. The Markov Chain will still sample from the correct distribution. However, it may take a longer time.

In practice this method works well. In Figure 4.4 I show a typical MCMC run. The high initial proposal variance quickly converges and the cumulative acceptance rate assymptotes at the desired value of 0.44. Slice sampling has replaced Metropolis-Hastings in many MCMC samplers used for population genetics such as in Beast (Drummond and Rambaut, 2007) and Phycas (Lewis et al., 2008). However, slice

Table 4.1: Accuracy of maximum likelihood inference estimating migration rates and population sizes. 1000 simulations were replicated at each migration rate for a symmetric two population model.

Estimates and Quantiles	4Nm		
	0.1	1	10
Average θ MLE(truth is 0.04)	0.048	0.043	0.047
Median θ MLE	0.045	0.041	0.045
Average $M = m/u$(truth is 2.5,25 and 250)	10.265	31.874	365.95
Median $M = m/u$	4.4480	19.845	237.67
percentage of time θ was within 95% confidence interval	86%	91%	85%
percentage of time M was within 95% confidence interval	81%	92%	86%

sampling can be expensive. Even though a slice sampler will never reject and will always sample a new value, a new sample requires at *minimum* 3 new function evaluations. Whereas with a properly tuned Metropolis-Hastings chain will on *average* require 2.27 function evaluations for a new sample.

4.4 Results

In the instance that a genealogy is precisely defined by some data, it is be possible to infer the maximum likelihood estimates of all of the population parameters(Felsenstein, 1992). I have done exactly this. Figure 4.5 shows an example of likelihood plots for a few parameters on a given tree. Table 4.1 shows the results of computing likelihood confidence intervals and checking how often the simulated value is within these confidence parameters.

Although the previous analysis is useful, it is rare for DNA sequences in the same species to be very informative in regards to a genealogy. There is rarely enough time for mutations to accumulate and the stretches of DNA are not long enough. Even if they were longer recombination would be an issue. For this reason the uncertainty in the estimate of a genealogy needs to be taken into account.

4.4.1 Simulated Sequences Comparison

I have further expanded the TPSC program to use Adaptive Metropolis Hastings and create posterior probabilities of the parameters in question. In order to test this method and program I have simulated data from a model that includes two equally

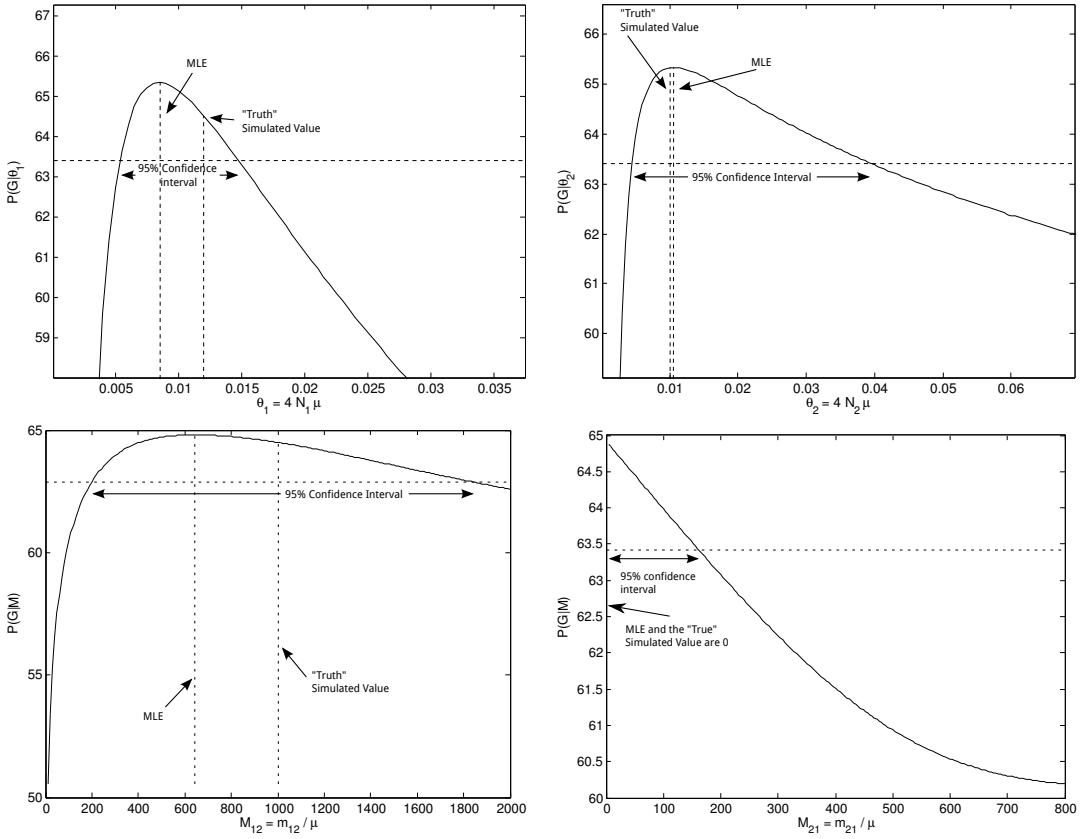


Figure 4.5: Plots of profile likelihood curves. Data was simulated from a two population model with migration in one direction. Labels are shown for simulated values, the maximum likelihood estimate of that value and the 95% confidence interval.

Table 4.2: Coverage of TPSC and MIGRATE. Fraction of the time that the true values Θ_T and M_T that were used to simulate the data were within the 95% credibility interval. For each Θ_T, M_T pair, 100 simulations were performed.

Program			Coverage of Θ			Coverage of M		
			Θ_T			Θ_T		
			0.001	0.01	0.1	0.001	0.01	0.1
TPSC	M_T	10	0.93	0.92	0.95	0.54	0.93	0.97
		100	0.9	0.88	0.97	0.94	0.99	0.96
		1000	0.97	0.94	1	0.97	0.97	1
MIGRATE	M_T	10	0.96	0.99	0.96	0.85	0.91	0.85
		100	0.97	0.95	0.97	0.88	0.98	0.63
		1000	0.97	0.96	0.87	0.92	0.81	0.5

sized populations sharing migrants symmetrically (migration rate is same in both directions). The scaled population size parameter θ takes values of 0.001, 0.01, and 0.1; while the migration parameter $M = m/\mu$ takes values of 10, 100 and 1000. For each scenario I simulated 100 replicates.

The TPSC program then inferred the Bayesian posteriors for each of the parameters. This program does not take into account for symmetry in the model thus while there are technically only two parameters in the model: θ and M, there are two observations of each parameter. One for each population. In addition the program MIGRATE which uses the older event based method also inferred the Bayesian posterior distributions.

Table 4.2 summarizes the results. TPSC as expected does poorly when the migration rate is low. Further investigation revealed that with low migration rate, TPSC under estimates the migration rate. Thus, it is possible to determine weather or not a dataset is appropriate for TPSC. When the migration rate rises TPSC does a good job of estimating the parameters. There is one *caveat*, TPSC is too certain about its estimates of high migration rate. This could mean that the posteriors are too broad. On the other hand, MIGRATE does a good job of estimating small migration rates. Large migration rates are poorly estimated by MIGRATE. The coverage becomes low and the program slows down.

These two methods could be used together. When migration rate is low, the event

based approach is not problematic and gives good results. When the migration rate is high, the new continuous method works better.

CHAPTER 5

NEW ORDER SPECIATION

5.1 Motivation

Previous to this chapter all the models have focused on one species in each analysis. For example, the chapter on model comparison was about choosing from competing models for one species exchanging migrants. Likewise the TPSC method has allowed for inference of high migration rates between populations of the same species. This section will expand the inference to multiple, but still closely related species.

Hey and Nielsen (2007) have pioneered much of the computational work for the isolation with migration(IM) models of speciation. They modeled the process in the following way. Looking backwards in time, all individuals in the same species can coalesce, while those in that are different species can not. There is a small rate of migration, this models horizontal gene transfer. Depending on the species horizontal transfer can be important (e.g. bacteria) or infrequent (e.g. mammals). Two closely related species can exchange genetic information, for example, dogs and wolves. These are widely recognized as different species, however, they will occasionally interbreed. Other examples include brown bears and polar bears and homo sapiens and Neanderthals(Green et al., 2010).

The IM model, specifies that at time t a speciation occurs. Looking backwards in time this means that all of a sudden every lineage is free to coalesce with another lineage. Since the speciation is abrubt, this is likely to be a good model for speciation occurring during sudden geological events. However, this model may not be good for speciation that happens more slowly for example sympatric speciation. This type of speciation is known to happen without any geographic separation of the gene flow, such as in abalone (Swanson et al., 2001).

Instead, what is proposed is a model that allows gradual speciation. This was first proposed by Peter Beerli (personal communication, 2009), however the mathematics and implementation outlined here is novel. In this case we are modeling two species that have recently speciated. Each sampled individual is a member of one of these two species. The lineages are modeled backwards in time and a lineage can be one of the present species or the parent species. Like Hey's model all individuals in the same species can interbreed, however unlike before there is not a single time that causes lineages in A and B to freely interbreed. Instead there is a distribution on speciation time(e.g. we have used gamma, and truncated normal). Each lineage individually crosses over to the parent species at it's own speciation time drawn from the speciation disribution. Lineages in the parent species can coalesce. This process continues until there is only one lineage.

This new model has the advantage that it allows for slow speciation. Instead of suddenly moving all lineages to the ancestral species, the lineages change to the ancestral species is parameterized. It is this parameterization that will allow the speciation to be described. For example, if we are using a truncated normal with a large mean and a large standard deviation, one could say that the speciation happened slowly, a long time in the past.

5.2 Methods

5.2.1 Hazard Functions

The probabilities in this model can be calculated using hazard functions. In the structured coalescent when looking backward in time coalescence events and migration events can happen. Combining these probabilities in order to create one probability density for events has not been difficult because both these events happen at an exponential rate. However in this model speciation events happen at non exponential rates.

A hazard function takes the following form

$$\lambda(t) = \frac{f(t)}{1 - F(t)}, \tag{5.1}$$

where $f(t)$ is the probability density function (pdf) of the time until an event happens; and $F(T)$ is the cumulative density function (cdf).

To calculate the cdf of any λ function, we use a method similar to the one used in when computing continuous migration event probabilities.

$$f(t) = \lambda(t) e^{-\int_0^t \lambda(x) dx} \tag{5.2}$$

For the new speciation method we have used a truncated Normal. However Gamma and a Log Normal functions can also be used to parameterize the speciation time. Below are their hazard functions:

Normal,

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-u)^2}{2\sigma^2}} \tag{5.3}$$

$$\lambda_{\mathcal{N}(\mu,\sigma)}(t) = \frac{\sqrt{\frac{2}{\pi}} e^{-\frac{(\mu-t)^2}{2\sigma^2}}}{\sigma\left(\operatorname{erf}\left(\frac{\mu-t}{\sqrt{2}\sigma}\right)+1\right)}, \tag{5.4}$$

Gamma,

$$f(x) = \frac{1}{\Gamma(k)\theta^k} x^{k-1} e^{-\frac{x}{\theta}} \tag{5.5}$$

$$\lambda_{\Gamma(\alpha,\beta)}(t) = \frac{\beta^{-\alpha} t^{\alpha-1} e^{-\frac{t}{\beta}}}{\Gamma\left(\alpha, \frac{t}{\beta}\right)} \tag{5.6}$$

Log Normal,

$$f(x) = \frac{1}{x\sqrt{q\pi\sigma^2}} e^{-\frac{(\ln x - u)^2}{2\sigma^2}} \tag{5.7}$$

$$\lambda_{\log\mathcal{N}(\mu,\sigma)}(t) = \frac{\sqrt{\frac{2}{\pi}} e^{-\frac{(\mu-\log(t))^2}{2\sigma^2}}}{\sigma t\left(\operatorname{erf}\left(\frac{\mu-\log(t)}{\sqrt{2}\sigma}\right)+1\right)} \tag{5.8}$$

In the above expressions erf is the Gauss error function and Γ is the gamma function. Likewise the integral of these hazard functions can be computed.

$$\int_{t_0}^{t_1} \lambda(t)_{\mathcal{N}(\mu,\sigma)} dt = \log\left(\operatorname{erf}\left(\frac{\mu - t_0}{\sqrt{2}\sigma}\right)+1\right) - \log\left(\operatorname{erf}\left(\frac{\mu - t_1}{\sqrt{2}\sigma}\right)+1\right), \tag{5.9}$$

$$\int_{t_0}^{t_1} \lambda(t)_{\Gamma(\alpha,\beta)} dt = \left(\frac{1}{\beta}\right)^{-\alpha} \beta^{-\alpha}\left[\log\Gamma\left(\alpha, \frac{t_0}{\beta}\right) - \log\Gamma\left(\alpha, \frac{t_1}{\beta}\right)\right], \tag{5.10}$$

and

$$\int_{t_0}^{t_1} \lambda(t)_{\log\mathcal{N}(\mu,\sigma)} dt = \log\left(\text{erf}\left(\frac{\mu - \log(t_0)}{\sqrt{2}\sigma}\right) + 1\right) - \log\left(\text{erf}\left(\frac{\mu - \log(t_1)}{\sqrt{2}\sigma}\right) + 1\right) \tag{5.11}$$

5.2.2 Tree-Likelihood Calculations

In order to make use of the hazard function, we have to be able to calculate the probability of a tree given a model and it's parameters. Before the probability density of a tree was calculated as the product of the probability density of all the events on the tree.

$$P(G) = \prod_{i=0}^{I} P(E_i) \tag{5.12}$$

Here G is a tree/genealogy, i is the number of the event and $P(E_i)$ is the probability of that event. For a coalescent event this is

$$P(E_i) = \frac{k(k-1)}{\theta} e^{\int_0^t \lambda_T dt} \tag{5.13}$$

Here λ_T is the total rate of events on the tree. Since λ_T is a constant this exponent is simply $\lambda_T t$. The probability of a migration event is

$$P(E_i) = m e^{\int_0^t \lambda_T t} \tag{5.14}$$

Here m_{ij} is the rate of the migration event i. The total rate λ_T is

$$\lambda_T = \sum_{i=1}^{N} \frac{k_i(k_i-1)}{\theta_i} + \sum_{i=1}^{N} \sum_{j=1, i\neq j}^{N} m_{ij} k_i \tag{5.15}$$

Here N is the number of populations and k_i is the number of lineages in population i. m_{ij} represents the immigration rate from population i to j.

The probability of a speciation event is

$$P(E_i) = \lambda_{sd} e^{\int_0^t \lambda_T t} \tag{5.16}$$

Table 5.1: The results of simulating 100 runs with different mean and standard deviation speciation time parameters.

		Average inferred mean	Average Inferred Standard Deviation
$\mu = 500$	$\sigma = 2$	500.8	2.32
	20	507.7	23.3
	200	578.7	280
$\mu = 1000$	$\sigma = 2$	1000.6	2.12
	20	1005.6	17.7
	200	1087.2	223
$\mu = 2000$	$\sigma = 2$	2000.4	1.84
	20	2003.2	17.7
	200	2056.2	172

Here *sd* is the speciation distribution we are using in the model. and the new λ_T is now

$$\lambda_T(t) = \sum_{i=1}^{N} \frac{k_i(k_i - 1)}{\theta_i} + \sum_{i=1}^{N} \sum_{j=1, i \neq j}^{N} m_{ij} k_i + \lambda_{sd}(t) \tag{5.17}$$

This new lambda has two parts, the old part that is constant and the new part that contains a hazard function. The hazard functions are integrated as shown in the preceding section while the constant parts come out as before.

5.3 Simulation Results

Haleh Ashki wrote a simulator that is able to simulate trees that use this model of inference. I simulated 100 different trees with 20 individuals from two populations. Each population had a population size of 1000. These populations exchange migrants and are a result of an ancestral population splitting in two. The speciation event occurred on average 2000,1000 or 500 generations ago. In order to test this model I varied the standard deviation of the Normal distribution used to simulate speciation events. I used 2,20 and 200 as the standard deviation.

In addition I have written a program that calculates the likelihood of the parameters used to infer a tree. It then proceeds to maximize those parameters in order to calculate an maximum likelihood estimate. I have estimated the mean and standard deviation of the truncated normal distribution used in the simulator. Table 5.1 show

the result of these runs. The data points show the values inferred by the program.

CHAPTER 6

CONCLUDING REMARKS

This dissertation has included three projects. The first project (Chapter 3) discusses improvement to Bayes factor-based model comparison. I developed a theorem and proof that allows to calculate the marginal likelihoods for independent data in separate computers and then combine them later. This has allowed the use of high performance computing which relies on parallel code. The rate of citations for this paper constantly increasing. Peter Beerli(personal communivation 2012) has started using Bayes Factors as a way to break the long stretches of genome into non recombinant units.

The second project improves the discrete structured coalescence with the transition probability based structured coalescence that replaces the discrete migration events with a continous approximation. It is an efficient method for the calculation of large migration rates. This allows the inference of high migration rates. In addition the complexity of the problem space is reduced. Instead of metropolizing over many different evens between many different populations a one dimensional numerical integral is used instead.

The third project was based on a rough idea of Peter Beerli proposing to treat speciation events that separate multiple samples from each of two species differently than the currently used methods. These methods subject all lineages at the same time to change the species label. I developed the formulae and methods to implement his idea into a working program that is described in this last chapter.

6.1 Population Genetics as a multi disciplinary field

Population genetics has been a very multi disciplinary field. The basis for most of the I have had to use tools from the fields of Statistics and computing to make inferences in Biology. It is my hope that I was able to add something useful to the field of population Genetics and my belief that I have. According to Google Scholar, the Bayes Factor paper has already been cited 63 times as of February 2013. The subject matter has ranged from mostly theoretical such as the speciation paper to very much applied in the Bayes Factor paper and in work with those outside the department.

BIBLIOGRAPHY

Beaumont, M., 1999. Detecting population expansion and decline using microsatellites. Genetics 153:2013–2029. 2.1.1

Bedford, T., S. Cobey, P. Beerli, and M. Pascual, 2010. Global migration dynamics underlie evolution and persistence of human influenza a (h3n2). PLoS Pathog 6. 1.1

Beerli, P., 1998. Estimation of migration rates and population sizes in geographically structured populations. Pp. 39–53, *in* G. R. Carvalho, ed. Advances in Molecular Ecology, *NATO sciences series, Series A: Life sciences*, vol. 306. ISO Press, Amsterdam. (document), 2.1.1, 2.3, 2.4.1, 4.3

———, 2004. Effect of unsampled populations on the estimation of population sizes and migration rates between sampled populations. Molecular Ecology 13:827–836. 3.3

———, 2006. Comparison of Bayesian and maximum likelihood inference of population genetic parameters. Bioinformatics 22:341–345. 3.2, 3.4, 3.4

Beerli, P. and M. Palczewski, 2010. Unified framework to evaluate panmixia and miration direction among multiple sampling locations. Genetics 185:313–326. 1.3.1, 2.7, 3.3.2, 3.4, 3.5

Camin, J. and R. Sokal, 1965. A method for deducing branching sequences in phylogeny. Evolution 19:311–326. 2.5

Carling, M. D. and R. T. Brumfield, 2007. Gene sampling strategies for multi-locus population estimates of genetic diversity (θ). PLoS One 2:160. 3.2

Drummond, A. J. and A. Rambaut, 2007. Beast: Bayesian evolutionary analysis by sampling trees. BMC evolutionary biology 7:214. 4.3

Fan, Y., R. Wu, M.-H. Chen, L. Kuo, and P. O. Lewis, 2011. Choosing among partition models in Bayesian phylogenetics. Molecular Biology and Evolution 28:523–532. 3.4, 3.5

Felsenstein, J., 1978a. Cases in which parsimony or compatibility methods will be positively misleading. Syst Zool 27:401–410. 2.5

———, 1978b. Theoretical Evolutionary Genetics. 2009 edition ed. Distributed by the Author, Seattle, WA. (document), 2.3, 2.4

———, 1981. Evolutionary trees from gene frequencies and quantitative characters: Finding maximum likelihood estimates. Evolution . 2.5, 2.5.1, 2.5.1

———, 1988. Phylogenies from molecular sequences: Inference and reliability. Annual Review of Genetics 22:521–565. 2.1.1

———, 1989. Phylip - phylogeny inference package (version 3.2). Cladistics 5:164–166. (document), 2.1, 2.5

———, 1992. Estimating effective population size from samples of sequences: inefficiency of pairwise and segregating sites as compared to phylogenetic estimates. Genet Res 59:139–147. 4.4

———, 2004. Inferring Phylogenies. Sinauer Associates, Sunderland, Massachusetts. 1.2.1, 2.5.1, 2.5.1, 2.6.2

———, 2005. Accuracy of coalescent likelihood estimates: do we need more sites, more sequences, or more loci? Molecular Biology and Evolution 23:691–700. 3.2

Fisher, R., 1930. The Genetical Theory of Natural Selection. Second edition, Dover, New York 1958 ed. Clarendon Press, Oxford, U.K. 2.3

Fitch, W. and E. Margoliash, 1967. Construction of phylogenetic trees. Science 115:279–284. 2.5

Friel, N. and A. Pettitt, 2008. Marginal likelihood estimation via power posteriors. Journal of the Royal Statistical Society: Series B Pp. 589–607. 3.5

Gelman, A., G. Roberts, and W. Gilks, 1996. Efficient Metropolis jumping hules. Bayesian statistics 5:599–608. 4.3

Geyer, C. J., 1991. Markov chain Monte Carlo maximum likelihood. *in* Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface, Pp. 156–163. Interface Foundation, Fairfax Station. 2.7

Green, R. E., J. Krause, A. W. Briggs, T. Maricic, U. Stenzel, M. Kircher, N. Patterson, H. Li, W. Zhai, M. H.-Y. Fritz, et al., 2010. A draft sequence of the neandertal genome. science 328:710–722. 5.1

Hasegawa, M., H. Kishino, and T. Yano, 1985. Dating the human-ape splitting by a molecular clock of mitochondrial dna. Journal of Molecular Evolution 22:160–174. 2.5.1

Hastings, W. K., 1970. Monte Carlo sampling methods using Markov chains and their applications. biometrika 57:97–109. 2.6.1

Hey, J., 2010. The divergence of chimpanzee species and subspecies as revealed in multipopulation isolation-with-migration analyses. Molecular Biology and Evolution 27:921–933. 1.3.3

Hey, J. and R. Nielsen, 2007. Integration within the Felsenstein equation for improved Markov chain Monte Carlo methods in population genetics. Proceedings of the National Academy of Sciences 104:2785–2790. 2.1.1, 5.1

Jeffreys, H., 1961. Theory of Probability. 3rd ed. ed. Oxford University Press, Oxford, U.K. (document), 2.7, 2.2, 3.1, 3.3

Jue, N., C. Koenig, and F. C. Coleman, in prep. Widespread genetic variability and the paradox of effective population size in the gag, *Mycterperca microlepis*, along the west florida shelf. . 1.1

Jukes, T. and C. Cantor, 1969. Evolution of protein molecules. Mammalian protein metabolism Pp. 21–123. 2.5.1

Kass, R. E. and A. E. Raftery, 1995. Bayes factors. Journal of the American Statistical Association 90:773–795. 3.4

Kimura, M., 1980. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. Journal of Molecular Evolution 16:111–120. 2.5.1

Kuhner, M., 2006. Lamarc 2.0: maximum likelihood and Bayesian estimation of population parameters. Bioinformatics 22:768–70. 3.2

Kuhner, M., J. Yamato, and J. Felsenstein, 1995. Estimating effective population size and mutation rate from sequence data using metropolis-hastings sampling. Genetics 140:1421–1430. 2.1.1

Lartillot, N. and H. Philippe, 2006. Computing Bayes factors using thermodynamic integration. Systematic Biology 55:195–207. 2.7

Lewis, P., M. Holder, and D. Swofford, 2008. Phycas: software for phylogenetic analysis. Storrs, CT: University of Connecticut. See www. phycas. org . 4.3

Metropolis, N., A. W. Rosenbluth, N. Rosenbluth, A. H. Teller, and E. Teller, 1953. Equation of state calculation by fast computing machines. Journal of Chemical Physics 21:1087–1092. 2.6.1, 3.3

Moler, C. and C. V. Loan, 2003. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. SIAM Review 45:3–49. 2.6

Neal, R. M., 2003. Slice sampling. The Annals of Statistics 31:705–767. 2.6.1

———, 2008. The harmonic mean of the likelihood: Worst Monte Carlo method ever. http://radfordneal.wordpress.com/2008/08/17/the-harmonic-mean-of-the-likelihood-worst-monte-carlo-method-ever/. 2.7, 3.4

Newton, M. A. and A. E. Raftery, 1994. Approximate Bayesian inference by the weighted likelihood bootstrap (with discussion). Journal of the Royal Statistical Society, Series B 56:3–48. 2.7

Palczewski, M. and P. Beerli, 2012. Population Model Comparison Using Multi-Locus Datasets. accepted for publication. 1.3.1

Pluzhnikov, A. and P. Donnelly, 1996. Optimal sequencing strategies for surveying molecular genetic diversity. Genetics 144:1247–1262. 3.2

Rannala, B. and Z. Yang, 1996. Probability distribution of molecular evolutionary trees: A new method of phylogenetic inference. Journal of Molecular Evolution 43:304–311. 2.5

Roberts, G. O. and J. S. Rosenthal, 2009. Examples of adaptive MCMC. Journal of Computational and Graphical Statistics 18:349–367. 4.3, 4.3

Roman, J. and S. R. Palumbi, 2003. Whales before whaling in the north Atlantic. Science 301:508–510. 1.1

Strobeck, C., 1987. Average number of nucleotide differences in a sample from a single subpopulation: A test for population subdivision. Genetics 117:149–153. 2.3.1, 2.4.1, 4.2

Swanson, W. J., C. F. Aquadro, and V. D. Vacquier, 2001. Polymorphism in abalone fertilization proteins is consistent with the neutral evolution of the egg's receptor for lysin (verl) and positive darwinian selection of sperm lysin. Molecular Biology and Evolution 18:376–383. 5.1

Swofford, D., 2003. PAUP*. Phylogenetic Analysis Using Parsimony (*and Other Methods). Version 4. Sinauer Associates, Sunderland, Massachusetts. 2.5

Swofford, D. and G. Olsen, 1990. Molecular Systematics, chap. 11, Pp. 411–501. Sinauer Associates, Sunderland, Massachusetts. 2.5.1

Takahata, N., 1988. The coalescent in two partially isolated diffusion populations. Genetical Research 52:213–222. 2.4.1, 4.2

Wakeley, J., 2008. Coalescent Theory: An Introduction. 1st edition ed. Roberts & Company Publishers, Greenwood Village, Colorado, USA. 2.3.1

Wilson, I. and D. Balding, 1998. Genealogical inference from microsatellite data. Genetics 150:499–510. 2.1.1

Wright, S., 1931. Evolution in mendelian populations. Genetics 16:97–159. 2.3, 2.4.2

———, 1938. Size of population and breeding structure in relation to evolution. Science 87:430–431. 2.4.2

Xie, W., P. O. Lewis, Y. Fan, L. Kuo, and M.-H. Chen, 2011. Improving marginal likelihood estimation for Bayesian phylogenetic model selection. Systematic Biology 60:150–160. 3.4, 3.5

BIOGRAPHICAL SKETCH

The author was born in Wrolaw, Poland in 1979. After moving to the United States in 1987 the author finished High School in 1998 at Roosevelt High School in Seattle, WA. The author graduated from the University of Washington, Seattle campus, with a degree in Applied and Computational Mathematical Science in 2002. The author started graduate school at FSU in 2005.