Data Driven Model Selection and Parameter Estimation Using Semi-Automatic Approximate Bayesian Computation to Reconstruct Population Dynamics From Ancient DNA

Adam Benjamin Rohrlach

Thesis submitted for the degree of

Master of Philosophy

in

Applied Mathematics

at

The University of Adelaide

(Faculty of Engineering, Computer and Mathematical Sciences)

School of Mathematical Sciences

May 8, 2014

Contents

Abstract

Population genetics is a discipline within the biological sciences that is concerned with the change in frequency of types of individuals in a population due to natural selection, mutation, genetic drift and gene flow. Genetic drift is the part of this process explained by random sampling. Important to the process of genetic drift is population structure and so we focus on the recovery of population sizes over time, given a set of DNA sequences.

With recent advances in computational power and a growth in the amount of data available, increasingly powerful techniques are being developed for the study of sequence data. Key advances in the early 1980's centred around 'the coalescent', a continuous time approximation to the Wright-Fisher model of reproduction, and these advances resulted in Skyline Plot methods for recovering population size estimates over time. Skyline Plots suffer from large variances for the 'coalescent' event times, and sources of error common to DNA sequence sampling schemes.

Approximate Bayesian Computation (ABC) is a class of likelihood-free methods for statistical inference. ABC techniques can trace their genesis back to the biological sciences due to the complexity of the models for reproduction (and hence the intractability of likelihood calculations). Unfortunately, like Skyline Plots, ABC also suffers from many sources of error, not least of which occurs when we can not use sufficient summary statistics.

To considerably reduce the effect of the error related with the use of insufficient summary statistics, we explore a process of semi-automatic summary statistic cal-

culation through the use of 'training data' (simulated under the coalescent model). We obtain a training set of data, and fit a linear model (under a Box-Cox transformation) for each parameter of interest, using common summary statistics for DNA sequences as predictor variables. We call these linear combinations of (insufficient) summary statistics the semi-automatic summary statistics, and using a new set of simulations, we perform ABC where a simulation is retained if the predicted parameter values are 'close enough' to the predicted parameters for the observed data. We analyse three sets of coalescent simulated data from three population models; the Constant, Exponential and Migration Models, and compare our findings with the corresponding Skyline Plot analyses performed in BEAST.

When we simulate data for training our linear model, we must specify a model of population size dynamics, and we explore methods to select a population model, given our data. A common means of model comparison used with ABC analyses is called Bayes Factors. We show that Bayes Factors perform poorly for our data, and highlight a fundamental bias inherent in any model comparison where the probability of a model, given an observed summary statistic, is employed. As an alternative to Bayes Factors, we apply multiple logistic regression (MLR) to classify our observed data into one of a candidate set of possible models. In conjunction with the MLR analysis, we use principal component analysis for visualisation, and introduce a method for attempting to identify when the correct model is not in the candidate model set, or when a classification seems reasonable. We show that this method of classification performs well for the three observed data sets using sensitivity analysis.

Due to the early stage of development of our work, we can not use real world data, and so we use a different type of simulation since our method uses coalescent simulations to train the model. We obtain sequence data simulated under a 'forward simulation' framework, a type of sequence simulation that looks forward in time. We define a two-step process for analysis that begins with MLR classification, and

then, under a model chosen by the MLR classification, uses semi-automatic summary statistic calculation for parameter estimation via ABC. We correctly identify this model of population dynamics, and perform parameter estimation on the data, comparing our results with the corresponding BEAST Skyline Plot analysis.

Signed Statement

I, Adam Rohrlach, certify that this work contains no material which has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition I certify that no part of this work will, in the future, be used in a submission for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree.

I consent to this copy of my thesis, when deposited in the University Library, being available for loan and photocopying, subject to the provisions of the Copyright Act 1968.

I also give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library catalogue and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

SIGNED: ..... DATE: .....

Dedication

I would like to dedicate my thesis to Allan and Mary Isobel Troughear.

You were the most wonderful and loving grandparents I could have hoped for. I hope I have made you as proud as you always seemed to be.

Acknowledgements

First, to my supervisors, Professor Nigel Bean and Dr Simon Tuke, thank you for seeing more in me from an early stage, than I saw in myself. It is because of your tireless encouragement, and small research projects during my undergraduate degree, that I decided to pursue post-graduate research. You are not only gifted lecturers, but two of the most patient and straight-faced mentors I could have hoped for. Our meetings have been so enjoyable, that I regularly forget that this is a job, and have begun to believe we might just take this show on the road.

To my Father, thank you for all of the advice these long years, the weekly lunches that allowed me to clear my mind, and the port. It seems you have always been waiting for me to get things right. Thank you for treating me no differently while I did not. To my Mother, thank you for the unconditional love, support, and encouragement. You have always made me feel as though nothing is beyond my reach. To my sisters, Prue and Paige, I thank you for the family meals where I was able to forget my work, and just enjoy being 'at home' again. In fact, to my entire family, I can not thank you enough for continuing to believe in me, even when all the available data suggested you should not.

To my two closest friends, Dan and Brett, thank you for coming on this five year long ride with me. It seems if I were ever in any danger of taking myself too seriously, you knew precisely when to step in. I have known you both for most of my life, and while this reflects poorly on the quality of company you keep, thank you for sticking with me during every phase of it.

To Professor Alan Cooper, and everyone at the Australian Centre for Ancient DNA, this project would not have been the same without you. Thank you for allowing me to sit in on countless meetings and answering every one of my questions. Specifically, to Julien and Oliver, thank you for taking the time to walk me through BEAST, and answering every email I sent you. I could not have done a single comparison of results without your help.

Finally, thank you to everyone in the School of Mathematical Sciences at the University of Adelaide. It has been an honour to work with all of you, and you have made the last two years a genuine pleasure. There are so many people I owe my thanks to, but I would like to thank a couple of you specifically. To David A, Heath, Jess, Kate, Nick, Nic and Stephen, thank you for all of the fun we have had, the help you have given me, and the countless coffees we have drunk over the last few years. I would be remiss not to mention three exceptional people, without whom I could not have completed this thesis with such fond memories.

To David, you have been a pleasure to work with, and I have learned so much about teaching since we 'teamed up'. Thank you for reading every word I have written, and never once berating me for them. To Mingmei, thank you for not only the last two years, but all of the time we have spent learning together. I doubt I could have succeeded without your influence, perspective, and strength. You are an amazing individual, and you pushed me harder than I would have pushed myself. To Vincent, I thank you for setting the bar so high. It is rare to meet someone so gifted, but still so easy to spend so much time with. Thank you for filling exam study periods with laughter, tea and talking british cakes.

Chapter 1

Introduction

Deoxyribonucleic acid (DNA) is a molecule that encodes all of the genetic information about a living organism. DNA can be represented by a sequence of the nucleotides A,C,G and T (adenine, cytosine, guanine and thymine), and a full sequence of information (a genome) can range from 1,759 to 150,000,000,000 nucleotides [23][32]. It is possible to focus on smaller subsections of the genome, that may perform some specific function say, called genes. An 'allele' is any one of the alternative forms a gene can take, and alleles are used to compare individuals within the same species.

Population genetics is a branch of the biological sciences that studies the frequency of alleles within a population with respect to the four main evolutionary processes:

Natural selection: The process by which alleles become more, or less, frequent in a population due to interaction with the environment.

Mutation: The process by which a nucleotide is changed. Causes of mutation include: molecular decay, replication error (when an individual's cells reproduce DNA incorrectly), DNA repair error, and mutagens (radiation for example).

Gene Flow: The process by which alleles move into and out of the 'gene pool' (the

alleles present in a population available for sampling) through migration.

Genetic Drift: The process by which allele frequencies alter due only to random sampling.

Important to the process of genetic drift is the size of the population of interest through time. We focus on the problem of recovering population size estimates given samples of DNA (sampled at different times in the population's history).

We begin with a review of the relevant literature. In Chapter 2 we present a description of a discrete-time model for species reproduction of non-overlapping generations, where each generation is of equal size, and mating is random (panmixia). We show how this model is generalised to a retrospective continuous-time approximation called the 'coalescent'. The coalescent derives its name from the fact that it models distinct lineages in a modern population (a lineage is a unique allele that has been sampled) finding pairwise common ancestors (coalescing) until a single most recent common ancestor (MRCA) is found. We further show how we can generalise this continuous-time approximation to allow for varying population sizes through time.

The coalescent allows us to draw a 'family tree' (called a genealogy) for the modern sequences, and this can be represented by a dendrogram. Given a set of observed sequences, and a mutation model (probabilities of nucleotide substitutions), we describe the method by which the likelihood of a tree is calculated. We then describe Felsenstein's method for finding the (heuristic) maximum likelihood tree for a set of sequences.

For a sequence alignment (from here referred to as a 'sample of sequences'), the maximum likelihood tree returns the most likely coalescent event times, and hence inter-coalescent event times. In Chapter 3 we describe a method of recovering population size estimates, called the Classical Skyline Plot, using these inter-coalescence times. Given the number of lineages present during the inter-coalescent

times, and a known rate of coalescence, we can estimate the harmonic mean population size in each interval. This method suffers from the over-fitting of parameters, and we describe a method called the Generalised Skyline Plot, which avoids this over-fitting, and also allows for sampling sequences in the past. Finally, the reasonable assumption of the correlation of population sizes through time culminates in the Bayesian Skyline Plot (BSP). These Skyline Plot methods all suffer from bias introduced by the rarely met assumption of panmixia, as geographical separation and natural selection commonly violate this assumption. Additionally, it is computationally intractable to sample every genealogy for a large sample of sequences, and hence maximum likelihood solutions are only locally maximal. Approximate Bayesian Computation is a likelihood free family of methods for sampling from posterior distributions when likelihood calculations are not sensible. We conclude Chapter 3 by introducing several common ABC sampling schemes, and discuss the desire for *sufficient* summary statistics in the case of sequence DNA. This chapter completes our summary of the existing work on the subject.

In Chapter 4 we introduce some of the most common summary statistics in DNA analysis and, due to the insufficient nature of the statistics, we describe a method for constructing the 'best' summary statistic from a subset of the statistics. We call the sampled sequences about which we wish to make inferences the 'ObsDat'. We simulate a large number of simulations from a grid for the parameters of interest, and we call this data set the 'TrainDat'. For each parameter of interest, we fit a linear model to the TrainDat where the summary statistics are the predictor variables, and the parameter of interest is the response variable (transformed under a Box-Cox transformation).

Next, we produce another set of simulations according to some prior distributions for the parameters of interest, and we call this the 'ABCDat'. For each simulation in the ABCDat, we calculate the 'predicted transformed parameter' value, and compare this to the 'predicted transformed parameter' for the ObsDat. We

then retain simulations where the predicted parameter values are 'close enough'. We compare our analysis with a corresponding BSP analysis for three population models: the Constant Model, where we have a single breeding population that remains at the same size going back in time; the Exponential Model, where a single breeding population decreases exponentially in size going back in time; and the so-called Migration Model, where four isolated populations of equal size remain at a constant size going back in time.

A key difference between our method of parameter estimation, and the BSP method of population estimate reconstruction is the initial model assumptions. BSP methods assume no population model, but do assume panmixia, and hence assumes that we have one united breeding population. Our method requires an assumed model of population dynamics for simulation to be defined before we estimate parameters. That is we must explicitly describe a population model before a simulation can be produced, and so the issue of model selection must be addressed.

In Chapter 5 we introduce a common method of post-hoc model comparison for ABC analyses called Bayes Factors. We show that for our analysis Bayes Factors perform poorly, and we highlight a fundamental bias in most analyses of this type. We suggest a method of model selection that is an example of supervised learning. Like our linear model for parameter estimation, we 'train' a multinomial logistic regression model (MLR) with the TrainDat for several candidate models. We define the MLR with the summary statistics as predictor variables, and the model as the response (outcome) variable. As a visual representation of the TrainDat, we perform a Principal Component Analysis on the data, and plot the first two principal components. We use the first two principal components of the ObsDat and evaluate the (normalised) distances between the ObsDat and the centroid for each model cluster, orthogonally projected onto the vector between these points. These distances allow for the identification of spurious classifications of the ObsDat. Our method performs well for identifying each of the three ObsDats, and

we present an algorithm that employs our method of data driven model selection, followed by our method of parameter estimation.

Due to the early stage of development of our work, we can not obtain sensible real world data, and so we must simulate a final data set. The final ObsDat we produce is from a different family of simulation methods to those used to produce our TrainDat and ABCDat, called 'forward simulation', and we begin Chapter 6 by describing this method. We apply our two-step method to this ObsDat, produced under a population model called the Bottleneck Model. We correctly identify the model of population dynamics, and we conclude Chapter 6 by presenting the results of our parameter estimation with a corresponding BSP analysis.

In Chapter 7 we conclude our findings and discuss possible extensions to the work presented in this thesis.

Chapter 2

The Coalescent Model and Approximation

2.1 Wright-Fisher Reproduction and the Coalescent Approximation

The 'coalescent model' is a modelling tool used to separate and describe the effects of genetic drift and mutation on the allelic frequencies for a population. It is a process that first builds a genealogy for the distinct alleles backward in time, before applying an independent mutational process forward in time. This method can be used to describe several different biological phenomena and allows for a computationally efficient method of simulating sequences for populations.

It is of importance to discuss the haploid number of a population as it will determine how we imagine a genealogy. The haploid number of a population is the number of copies of chromosomes in a gamete (sperm or egg).

A haploid organism has only one copy of each chromosome per cell, and individuals require only one 'parent'. A diploid organism contains two copies of each chromosome per somatic cell, and individuals require two parents. Bacteria is an

example of a haploid organism, while human beings are an example of a diploid population. We inherit one chromosome from our mother, and one chromosome from our father. When considering a haploid population, children have only one parent, and this model of genetic inheritance is still worth considering with respect to following a single parental line. In most species, including human beings, mitochondrial DNA (mtDNA) is inherited solely from the mother. When analysing mtDNA, the diploid population can now be treated as a haploid population since the father contributes nothing to mtDNA, hence individuals need only have one parent.

Haploid methods were the first models developed, and were the foundation for methods for analysing diploid populations, and populations with even greater ploidy. However, methods for haploid populations are still extremely useful. Unlike nuclear DNA, mtDNA only undergoes recombination with copies of itself, and is passed unchanged from parent to offspring. Because of this, and due to the higher rate of mutation, mtDNA is an extremely powerful tool for analysing populations (through the matrilineal line). A similar analysis using the non-recombining part of the Y-chromosome can follow a patrilineal line. In all of our following work, we shall only be considering the case of a haploid population.

To describe how a population evolves over many discrete generations, we must first begin with a description of how two consecutive generations interact. The coalescent model is based on the Wright-Fisher reproduction model which, for a population of breeding individuals, has the following assumptions;

- Each generation is non-overlapping.
- Each generation is of finite size N.
- There is random mating (panmixia) independent of the gene.

These assumptions ensure that generations do not interbreed, are of a constant size and no one allelic form is more likely to succeed than any other.

If we ignore mutation, we can model backwards in time from the present (labelled generation 0), allowing individuals to randomly select a parent in the previous generation. We repeat this process until a Most Recent Common Ancestor (MRCA), or the most modern individual that everyone in a sample can draw their ancestry back to, is found (see Figure 2.1).

One of the most computationally efficient ways of simulating this process, and generating a genealogical tree, is the Kingman's Coalescent [20]. Kingman's Coalescent was introduced in 1982, and is a continuous-time diffusion approximation of the discrete time model of Wright-Fisher Reproduction. From here we refer to Kingman's Coalescent as "The Standard Coalescent Model".

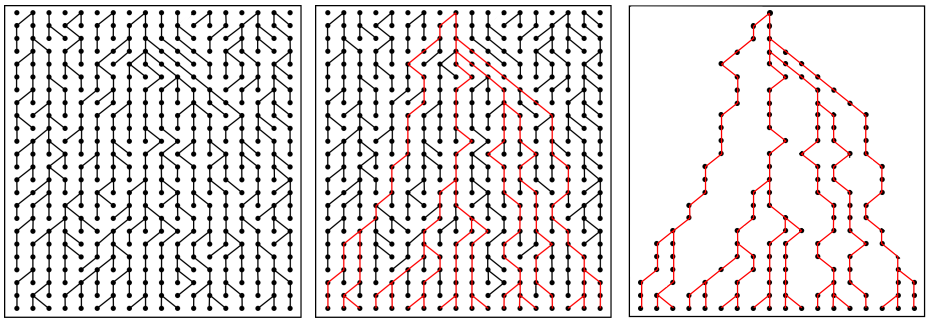


Figure 2.1: An example of a genealogy built 'backwards' in time (time moves from the past to the present down the page).

2.1.1 The Standard Coalescent Model

Consider k distinct lineages. If any two 'children' are to avoid selecting the same 'parent' in the previous generation the 'first' child may select any parent from the N parents. The next child may now select any parent from the previous generation except for the parent the first child has selected. Similarly, the i^{th} child may select any parent from the previous generation except for the $i-1$ parents selected by the previous $i-1$ children.

By this argument, the event that no two of the k 'children' will select the same

'parent' in the previous generation, and hence have their lineages coalesce, occurs with probability,

$$\prod_{i=0}^{k-1} \frac{N-i}{N} = \prod_{i=0}^{k-1} \left(1 - \frac{i}{N}\right)$$
$$= 1 - \frac{\binom{k}{2}}{N} + \mathcal{O}\left(\frac{1}{N^2}\right),$$

where the probability that more than two distinct lineages coalesce is $\mathcal{O}\left(\frac{1}{N^2}\right)$. That is, they are dominated by a $\frac{1}{N^2}$ term. Similarly, since children select parents independently with some probability $\mathcal{O}\left(\frac{1}{N}\right)$, the probability that two or more pairs of distinct lineages coalesce in the same generation will be $\mathcal{O}\left(\frac{1}{N^2}\right)$.

We rescale time such that N generations pass for every one unit of scaled time, and define $T(k)$ to be the scaled time until the first coalescent event, given that there are k distinct lineages, and see in the limit as $N \to \infty$, that

$$P(T(k) > t) = \lim_{N\to\infty} \left(1 - \frac{\binom{k}{2}}{N}\right)^{\lfloor Nt \rfloor} = e^{-\binom{k}{2}t}.$$

Hence, we have that $T(k)$ is, in the limit, exponentially distributed with rate $\binom{k}{2}$, and that the probability that more than two lineages coalesce is negligible.

Under the coalescent approximation, the number of distinct lineages in the ancestry of a finite population of size N decreases in steps of one back in time with $T(k)$ the time from k to $k-1$ lineages. That is, we can model the coalescent process as a random bifurcating tree, where the $N-1$ coalescence times, $T(N), T(N-1), \ldots, T(2)$ are mutually independent, exponentially distributed random variables with $T(k)$ having mean $\binom{k}{2}^{-1}$ for $k = 2, \ldots, N$. Note that the $N-1$ independent exponential inter-coalescent times can be easily generated using the R-function `rexp` [34]. An example of these times for five modern lineages, fitted to an arbitrary tree, can be seen in Figure 2.2.

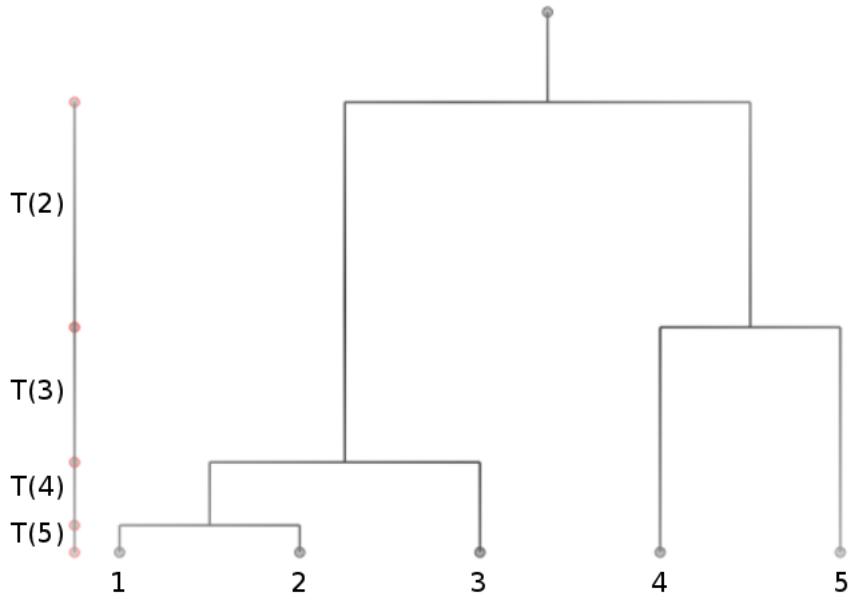


Figure 2.2: A realisation of a coalescent process with $n = 5$ modern samples and inter-coalescent times $T(k)$.

We can now define the time until the MRCA of a sample of n individuals as

$$T_{\text{MRCA}}(n) = \sum_{j=2}^{n} T(j).$$

It is worth noting that dramatically increasing the sample size does not dramatically increase the T_{MRCA}. To see this, first consider that since $T(j)$ is an exponential random variable with mean $\binom{j}{2}^{-1}$, we have that

$$E\left[T(j)\right] = 1/\binom{j}{2} = \frac{2}{j(j-1)},$$

and specifically,

$$E\left[T(2)\right] = 1/\binom{2}{2} = 1. \tag{2.1}$$

Hence,

$$\begin{aligned} E\left[T_{\mathrm{MRCA}}(n)\right] &= \sum_{j=2}^{n} E\left[T(j)\right] \\ &= \sum_{j=2}^{n} \frac{2}{j(j-1)} \\ &= 2\sum_{j=2}^{n} \left(\frac{1}{j-1} - \frac{1}{j}\right) \\ &= 2\left(1 - \frac{1}{n}\right). \end{aligned}$$

As we let the sample size $n \to \infty$,

$$\begin{aligned} \lim_{n\to\infty} E\left[T(n)\right] &= \lim_{n\to\infty} 2\left(1 - \frac{1}{n}\right) \\ &= 2 \\ &= 2E\left[T(2)\right] \quad \text{(from 2.1)}. \end{aligned}$$

We have that in the limit as the sample size $n \to \infty$, the T_{MRCA} is only twice as large as the the T_{MRCA} of a sample of size 2.

Under the assumptions of a single population, with non-overlapping generations of a constant fixed population size, we refer to the coalescent approximation as the Standard Coalescent Model.

2.1.2 Modifications to the Standard Coalescent Model

The assumptions made for the Standard Coalescent Model are somewhat unreasonable when applied to real world populations. We expect that populations sizes will vary over time. Similarly, it is reasonable to expect that populations might

be made up of several 'sub-populations' with restricted interaction between each other. We now describe some of these population models, and how they can be conveniently described in terms of the Standard Coalescent Model with a 'rescaling of time'.

Allowing for a Changing Population Size

If we wish to model the dynamics of a population with a varying effective population size (the number of breeding individuals), we must know how the population has varied over time *a priori*. Here we present the work introduced in 1994 by Griffiths *et. al.*[**?**]. Let $N_e(r)\ \forall\ r \in \mathbb{Z}^+$, be the known effective population size r generations in the past, and define $N_e(0) = N$. Define the relative size (to the most modern population) function to be

$$v_N(t) = \frac{N_e(\lfloor Nt \rfloor)}{N} = \frac{N_e(r)}{N},$$

where $\frac{r}{N} \leq t \leq \frac{r+1}{N}$ for $r \in \mathbb{N} \cup \{0\}$.

If every generation has a sufficiently large population size for the coalescent approximation to be reasonable, then

$$\lim_{N\to\infty} v_N(t) = v(t),\ \ t > 0,$$

exists. We call these the 'intensity functions', and we have that $v(t) > 0,\ \ t \geq 0$.

For example, if $N_e(r) = \left(1 - \frac{\beta}{N}\right)^r N$ (a geometric decay of rate $\frac{\beta}{N}$ in population size going back in time), and time is scaled in N units, then

$$\begin{aligned}\lim_{N\to\infty} v_N(t) &= \lim_{N\to\infty} \left(1 - \frac{\beta}{N}\right)^{\lfloor Nt \rfloor} \\ &= e^{-\beta t}.\end{aligned}$$

So, for a population undergoing exponential growth going forward in time, we define,

$$v(t) = e^{-\beta t}$$

where $t \geq 0$.

Similarly, for a population that undergoes a single bottleneck event where the population size changes from N to βN at time t_0, we find,

$$\lim_{N\to\infty} v_N(t) = \lim_{N\to\infty} \begin{cases} \frac{N}{N} = 1, & 0 \leq t < t_0, \\ \frac{\beta N}{N} = \beta, & t \geq t_0, \end{cases}$$

and define

$$v(t) = \begin{cases} 1, & 0 \leq t < t_0, \\ \beta, & t \geq t_0. \end{cases}$$

Note that this definition incorporates a constant population size model, by letting $t_0 = \infty$.

Now consider studying the evolution of the genetic structure of a sample of n individuals with some $v(t)$, which is a result of the limit as $N \to \infty$ of the Wright-Fisher reproduction model, and as before, we measure time in units of N generations.

If we let

$$\lambda(t) = \frac{1}{v(t)}, \quad t > 0,$$

we can describe the coalescent approximation with a varying population size with the following process.

Consider a process with a sample of n individuals taken at time $t = 0$. Let $\{A_n(t) : t \geq 0\}$ be the number of distinct lineages at time t in the past. $A_n(\cdot)$ is a Markov death process that starts at $A_n(0) = n$, and decreases in steps of one until $A_n(t) = 1$ (the MRCA is reached).

Intuitively, the probability of coalescence will increase for a smaller population, as individuals are selecting from fewer parental individuals. Since the time scale directly reflects the rate of coalescence, we allow this scaling to vary over time. We define $g(t)$ to be the amount of coalescence time that has passed after t (for some t) generations as

$$g(t) = \sum_{j=0}^{t} \frac{1}{N(j)}.$$

Therefore τ units of coalescence time ago corresponds to $g^{-1}(\tau)$ generations having passed. This function $g(t)$, and the inverse $g^{-1}(\tau)$, allow us to map between the non-linear time of the variable population size model, and the linear time of the standard coalescent model.

However, these functions can be well approximated using the intensity functions. We then have that t generations ago is approximately equivalent to

$$\Lambda(t) = \int_0^t \frac{1}{Nv(x)} dx, \;\; t > 0,$$

units of coalescence time (and that τ units of coalescence time ago corresponds to approximately $\Lambda^{-1}(\tau)$ generations having passed).

In effect, we 'stretch out' time when the population is larger, and 'shrink down' time when the population is smaller to account for the varying rate of coalescence. That is, a genealogy for a varying population can be generated by applying Λ^{-1} to a genealogy generated under the standard coalescent.

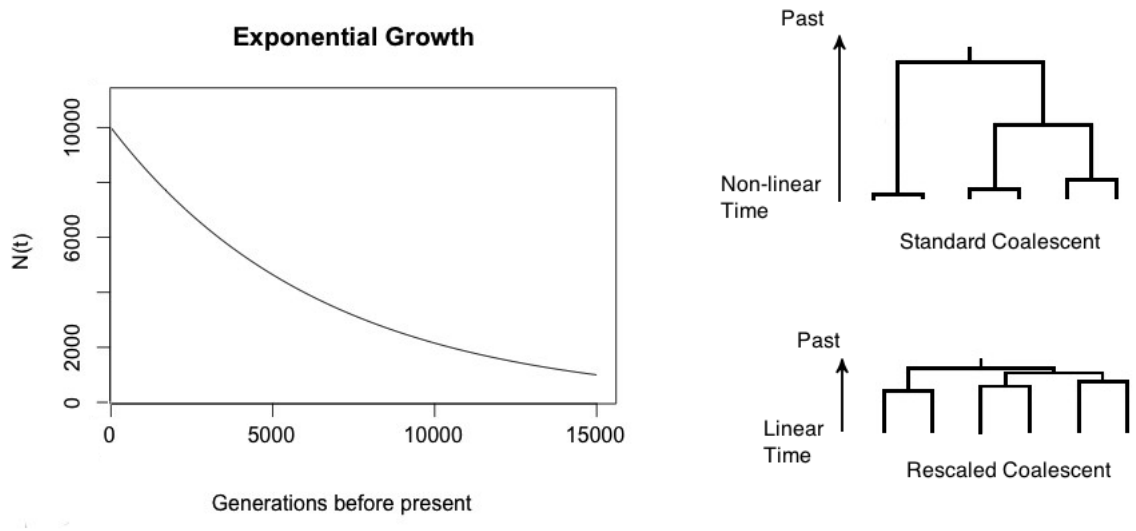


Figure 2.3: A rescaling of the Standard Coalescent Model for an exponentially growing population.

An example of this for a population undergoing exponential growth forward in time is given in Figure 2.3.

We have already shown that for this model

$$v(t) = e^{-\beta t}.$$

So we have that

$$\begin{aligned}\Lambda(t) &= \int_0^t \frac{1}{Ne^{-\beta s}} ds \\ &= \frac{1}{N}\left[\frac{e^{\beta s}}{\beta}\right]_0^t \\ &= \frac{1}{\beta N}\left(e^{\beta t} - 1\right),\end{aligned}$$

and hence, t generations ago corresponds to $\frac{1}{\beta N}\left(e^{\beta t} - 1\right)$ units of coalescence time having passed. Similarly, for

$$\tau = \frac{e^{\beta t} - 1}{\beta N},$$

we rearrange τ to find t, and find

$$\begin{aligned} (N\beta\tau + 1) &= e^{\beta t} \\ \implies \frac{1}{\beta}\ln(N\beta\tau + 1) &= t. \end{aligned}$$

So,

$$\Lambda^{-1}(\tau) = \frac{1}{\beta}\ln(N\beta\tau + 1),$$

and hence τ units of coalescence time ago corresponds to $\frac{1}{\beta}\ln(N\beta\tau + 1)$ generations having passed since present.

2.1.3 Population Structure

Populations can be separated geographically, and it is necessary to be able to introduce this idea into the framework of the coalescent model. Sensibly, if two sub-populations are separated by some geographical distance, but still have some probability of interacting and breeding, we expect that lineages within populations will most likely coalesce with one another. Similarly, since the population has been sub-divided, we also expect a faster rate of coalescence within the sub-populations due to their smaller population sizes, relative to the population as a whole. Interestingly, this 'structure' can also be applied to non-geographical structures, such as biological structures involving age, or allelic classes.

Consider a population of size N, as before, but sub-divided into patches of size $N_i,\ i \in \{1, \ldots, M\}$ such that $\sum_{i=1}^{M} N_i = N$. We make the same assumptions as for the Standard Coalescent Model, but we also assume that each individual in each generation produces (effectively) infinite offspring.

Let $m_{ij},\ i, j \in \{1, \ldots, M\}$ be the probability that one of the offspring in patch i 'migrates' to patch j. We can then define $b_{ij},\ i, j \in \{1, \ldots, M\}$ as the probability

that a randomly selected individual in patch i originated from patch j by

$$b_{ij} = \frac{N_j m_{ji}}{\sum_{k=1}^{M} N_k m_{ki}}.$$

Note that each of the N_i individuals in the previous generation in patch i was equally likely to have given rise to each of the $b_{ij}N_j$ individuals in patch j that originated from patch i. So we have that the number of offspring that a randomly selected individual in patch i can contribute to patch j in the next generation is binomially distributed with parameters N_j and $b_{ji}N_i^{-1}$. Hence the joint distribution of the numbers of offspring contributed to patch j in the next generation by all individuals in all patches in the previous generation will be multinomial.

By looking at the process going backwards in time as before, two lineages can coalesce if and only if they choose the same parental patch, and then the same parent within that patch. Hence, two lineages in patches i and j coalesce with probability $b_{ik}b_{jk}N_k^{-1}$.

It is possible to approximate this model by a continuous-time Markov process. However, as we allow $N \to \infty$ in the appropriately scaled time, we must decide how the parameters M, N_i and b_{ij} scale with it. Selecting how these parameters scale lead to two different biological descriptions of migration; weak and strong migration.

The key difference in assumptions between the two models of migration is whether or not the expected number of migration events per generation tends to infinity as the total number of organisms tends to infinity. If migration is not extremely common, the organism experiences 'weak migration', and the expected number of migrations per generation is finite. The coalescent behaviour of the distinct lineages will be dominated by the origin of their sub-population. However, if migration is extremely common, the population experiences 'strong migration'. In this case the initial distribution of lineages will have a diminished effect, and the expected number of migrations per generation is infinite.

The Structured Coalescent under Weak Migration

If we assume $M, c_i = N_i/N$ and $B_{ij} = 2Nb_{ij},\ i \neq j$ all remain constant as $N \to \infty$, we force both the probability of per generation coalescence and the probability of backward migration to be $\mathcal{O}(\frac{1}{N})$. By making this assumption, we allow the probability of a coalescent event to be comparable to the probability of a migration event (this will be seen to be of importance when we consider strong migration). We also have that the probability, per generation, of more than one lineage migrating, more than two lineages coalescing or that lineages migrate and coalesce are $\mathcal{O}(\frac{1}{N^2})$, or smaller.

In the limit as $N \to \infty$ we have only two possible events. Either two lineages coalesce, or a lineage migrates. Each pair of lineages in patch i coalesce independently at rate $\frac{1}{c_i}$, and each lineage in patch i migrates backwards in time independently to patch j at rate $\frac{B_{ij}}{2}$.

Since both types of events occur according to independent Poisson processes, the waiting time between either type of event occurring is exponentially distributed with rate

$$h(k_1, \ldots, k_M) = \sum_{i=1}^{M} \left[\frac{\binom{k_i}{2}}{c_i} + \sum_{j \neq i} k_i \frac{B_{ij}}{2} \right],$$

where k_i is the number of distinct lineages in patch i.

When an event occurs, it is a coalescence in patch i with probability

$$\frac{\binom{k_i}{2}/c_i}{h(k_1, \ldots, k_M)}.$$

Here a randomly selected pair of lineages in patch i coalesce, and k_i decreases by one.

An event could also be a migration from patch i to patch j, and this event occurs with probability

$$\frac{k_i B_{ij}/2}{h(k_1, \ldots, k_M)}.$$

Here a randomly selected lineage migrates from patch i to patch j, and k_i decreases by one, and k_j increases by one.

It is intuitive to imagine that under extremely weak migration ($b_{ij} << 1$), the reconstructed genealogy of a population will be strongly affected. That is, if the probability of migrating is very small, then the likelihood of the underlying genealogical tree will be quite different from that of a single breeding population, as some lineages within patches will not have access to one another for extended periods of time. If the probability of migration is very small, then lineages within patches are far more likely to coalesce before a rare migration coalescence occurs. Hence, $E[T_{\text{MRCA}}]$ and $\text{Var}(T_{\text{MRCA}})$ will be significantly different also (see Figure 2.4).

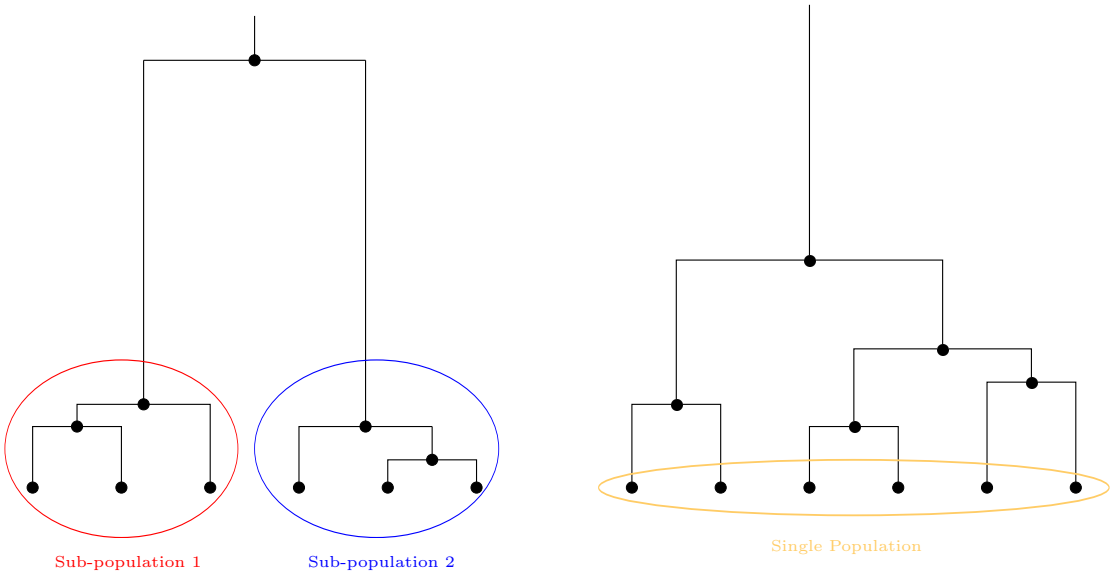


Figure 2.4: Genealogical trees for a weak migration model (left) where sub-populations coalesce first, and single breeding population (right) where all lineages coalesce equally likely.

The Structured Coalescent under Strong Migration

It seems intuitive that under strong migration, the underlying genealogical tree will be reasonably similar to if we had a single breeding group, since all individuals in the subpopulations have access to one another. Similarly it seems reasonable that $E[T_{\text{MRCA}}]$ and $\text{Var}(T_{\text{MRCA}})$ will also be comparable to that of a single breeding group. This is largely true, except that the scaling of time changes. While it is not true for weak migration, under strong migration we can again find a simple linear rescaling of time to relate the migration model to the Standard Coalescent Model.

Strong migration is characterised by claiming that the per generation migration probabilities are not $\mathcal{O}(\frac{1}{N})$, and hence

$$\lim_{N\to\infty} Nb_{ij} = \infty.$$

Since coalescent probabilities are still $\mathcal{O}(\frac{1}{N})$, we have that for large N, migration events are far more likely than coalescent events.

As $N \to \infty$, we effectively have infinitely many migration events occurring between coalescent events. Since coalescent events can only occur if two individuals select the same parent in the same patch, and migration occurs infinitely fast on a coalescent time scale, we use the stationary distribution of the migration matrix $B = [b_{ij}]$.

Assuming that B has a stationary distribution, let π_i be the stationary probability of a lineage being in patch i. Hence, a given pair of lineages will occupy patch i for π_i^2 proportion of the time. Since coalescence occurs in patch i at a rate $\frac{1}{c_i}$, the overall rate of coalescence is

$$\alpha := \sum_{i=1}^{M} \frac{\pi_i^2}{c_i}.$$

As usual, pairs of lineages coalesce independently and hence the total rate of coalescence when there are k distinct lineages is $\binom{k}{2}\alpha$. If time is rescaled in units of N/α, then we regain the Standard Coalescent Model.

2.1.4 Using the Coalescent Approximation

Once the genealogical tree has been created, the process of mutations on the sequences can now be added independently forward in time from the MRCA, with some pre-specified sequence. The process of mutations occurs independently on all branches according to a Poisson process with rate $\frac{\theta}{2}$, where $\theta = 2N\mu$ (for a haploid population), and μ is the per site mutation rate for an individual in any given generation.

Note that to perform a coalescent analysis, one must know the population size dynamics over time. Currently researchers go about obtaining $N(t)$ via Maximum Likelihood Methods for the underlying genealogy of a set of sequences, as described by Felsenstein in his seminal paper on the topic [14].

2.2 Felsenstein's Maximum Likelihood Methods for Evolutionary Trees

If we consider the evolutionary tree to be the item that we want to estimate, then we require a method to calculate the probability of observing a set of sequences given a particular tree. To achieve this, we need to find the probability of observing the sequence of elements S_1 change to the sequence of elements S_2 on a particular branch of a tree over time t (measured in any unit). If we assume the individual elements change independently, we can calculate the probabilities element-wise for the whole tree, and take the product of these probabilities [14]. Hence, we focus on calculating the probability for a single element. We assume these probabilities reflect a Markov process as the probability of a single element of the sequence changing depends only on its current condition, and not on its history.

Let $P_{ij}(t)$ be the probability that a lineage in state i will be in state j after t units of time, where $i, j = 1, \ldots, k$ and these correspond to k distinct possible lineage

states. For example, the distinct states may be A,G,C and T, the four base pairs of DNA base pairing.

Let $\pi_1, \ldots, \pi_k$ be the conditional probabilities of an element of the sequence being replaced by $1, \ldots, k$ respectively, given a replacement has occurred. Finally, over a time interval Δt, let the probability that an element is replaced be $u\Delta t$. Hence we have that,

$$P_{ij}(\Delta t) = (1 - u\Delta t)\delta_{ij} + u\Delta t\pi_j, \tag{2.2}$$

where δ_{ij} is the Kronecker delta function with

$$\delta_{ij} = \begin{cases} 0, & i \neq j, \\ 1, & i = j. \end{cases}$$

Since the process is Markovian, from equation (2.2) it can easily be shown that for arbitrary t

$$P_{ij}(t) = e^{-ut}\delta_{ij} + (1 - e^{-ut})\pi_j. \tag{2.3}$$

Next we introduce the idea of a 'genealogical tree'. A genealogical tree is a graphical representation of a genealogy, with (current) time starting at the bottom and going into the past as we go up. A 'node' is a point connecting two or more branches with the convention that if node i connects to node j from above, node i is the parent of node j, and node j is the child of node i. For example, in Figure 2.5, node 5 is the parent of node 1 and node 4 is a child of node 6.

Specifically, we have the following interpretations:

1. The 'leaves' (the nodes without children) represent individuals with sequenced data, and we will see that these need not be contemporaneous.

2. 'Parent nodes' (any node that is not a leaf) represent coalescence events, that is, points where lineages are no longer distinct.

3. Branch lengths represent time between coalescence events.

4. The 'root node' represents the MRCA of the genealogy.

For example, in Figure 2.5 we have leaves at nodes 1, 2, 3 and 4, parent nodes 5 and 6, the MRCA at node 0 (also a parent node) and branch lengths $v_1, \ldots, v_6$.

A 'tree topology' is the branching pattern of a tree. That is, a topology is concerned only with which nodes are connected to which nodes, not the the branch lengths connecting the nodes.

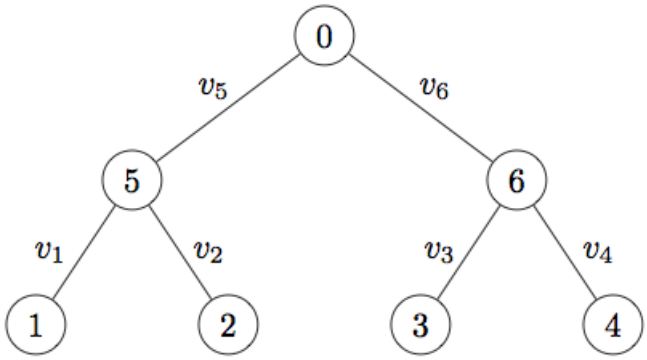


Figure 2.5: An example of a possible tree with nodes {0,1,...,6} and branch lengths v_1,...,v_6.

If we assume that after a coalescent event (forward in time) two lineages evolve independently, we can then calculate the likelihood of the tree as the product of the transitions from each child to parent node. So for the tree in Figure 2.5 we would have

$$L = \pi_0 P_{s_0 s_5}(v_5) P_{s_5 s_1}(v_1) P_{s_5 s_2}(v_2) P_{s_0 s_6}(v_6) P_{s_6 s_3}(v_3) P_{s_6 s_4}(v_4),$$

where s_i is the state at node i and π_0 is the probability of seeing state s_0.

In practice, only the leaves are known, so we need to sum over all possible values of the remaining nodes, hence

$$L = \sum_{s_0} \sum_{s_5} \sum_{s_6} \pi_0 P_{s_0 s_5}(v_5) P_{s_5 s_1}(v_1) P_{s_5 s_2}(v_2) P_{s_0 s_6}(v_6) P_{s_6 s_3}(v_3) P_{s_6 s_4}(v_4).$$

We can group the terms by the unobserved states and find

$$L = \sum_{s_0} \pi_0 \left\{ \sum_{s_5} P_{s_0 s_5}(v_5) \left[P_{s_5 s_1}(v_1) P_{s_5 s_2}(v_2) \right] \right\} \left\{ \sum_{s_6} P_{s_0 s_6}(v_6) \left[P_{s_6 s_3}(v_3) P_{s_6 s_4}(v_4) \right] \right\}$$

which matches the tree topology.

To begin to discuss simplifying the method by which we obtain a maximum likelihood tree, we must first discuss the notion of reversibility. If we can determine that our process is reversible we will be able consider our process of mutation happening either forward or backwards in time. This will allow for likelihoods to be calculated going up the tree, and then back down, without changing the likelihood of the tree. The reasons for this will become obvious shortly.

For a Markov process to be reversible we require that for all i,j and t

$$\pi_i P_{ij}(t) = \pi_j P_{ji}(t).$$

First, we note that

$$\pi_i \delta_{ij} = \pi_j \delta_{ji} = \begin{cases} 0, & i \neq j, \\ \pi_i & i = j. \end{cases} \tag{2.4}$$

Substituting equation (2.4) into equation (2.3) we have

$$\begin{aligned} \pi_i P_{ij}(t) &= \pi_i \delta_{ij} e^{-ut} + (1 - e^{-ut}) \pi_i \pi_j \\ &= \pi_j \delta_{ji} e^{-ut} + (1 - e^{-ut}) \pi_j \pi_i \\ &= \pi_j P_{ji}(t) \end{aligned}$$

which shows that the substitution rate is reversible.

We now define $L_s^{(k)}$ as the likelihood based on the data at or below node k on the tree, given we know that the state at node k is s. Note that if node k is a leaf then

$$L_s^{(k)} = \begin{cases} 0, & s \neq s_k, \\ 1, & s = s_k. \end{cases}$$

As we have a bifurcating tree, for each node k with immediately descended children i and j we get the relationship

$$L_{s_k}^{(k)} = \left(\sum_{s_i} P_{s_k s_i}(v_i) L_{s_i}^{(i)} \right) \left(\sum_{s_j} P_{s_k s_j}(v_j) L_{s_j}^{(j)} \right), \tag{2.5}$$

and beginning at the leaves we move up the tree, node by node, until we find

$$L = \sum_{s_0} \pi_0 L_{s_0}^{(0)}. \tag{2.6}$$

Applying equation (2.5) to equation (2.6), and letting $k = 0$ yields

$$L = \sum_{s_0} \pi_0 \left(\sum_{s_i} P_{s_0 s_i}(v_i) L_{s_i}^{(i)} \right) \left(\sum_{s_j} P_{s_0 s_j}(v_j) L_{s_j}^{(j)} \right).$$

Now using the property of reversibility we have

$$\begin{aligned} L &= \sum_{s_0} \left(\sum_{s_i} \pi_0 P_{s_0 s_i}(v_i) L_{s_i}^{(i)} \right) \left(\sum_{s_j} P_{s_0 s_j}(v_j) L_{s_j}^{(j)} \right) \\ &= \sum_{s_0} \left(\sum_{s_i} \pi_i P_{s_i s_0}(v_i) L_{s_i}^{(i)} \right) \left(\sum_{s_j} P_{s_0 s_j}(v_j) L_{s_j}^{(j)} \right) \\ &= \sum_{s_i} \sum_{s_j} \pi_i L_{s_i}^{(i)} L_{s_j}^{(j)} \left(\sum_{s_0} P_{s_i s_0}(v_i) P_{s_0 s_j}(v_j) \right), \end{aligned}$$

which by the Chapman-Kolmogorov equation gives

$$L = \sum_{s_i} \sum_{s_j} \pi_i L_{s_i}^{(i)} L_{s_j}^{(j)} P_{s_i s_j}(v_i + v_j). \tag{2.7}$$

Equation (2.7), and hence the likelihood of the tree, does not rely on the position of the root node, but only on the total length of the two branches, $v_i + v_j$.

Hence we can think of testing each rooted tree as testing the likelihood of a family of unrooted trees. For example, the tree in Figure 2.5 will have the same likelihood

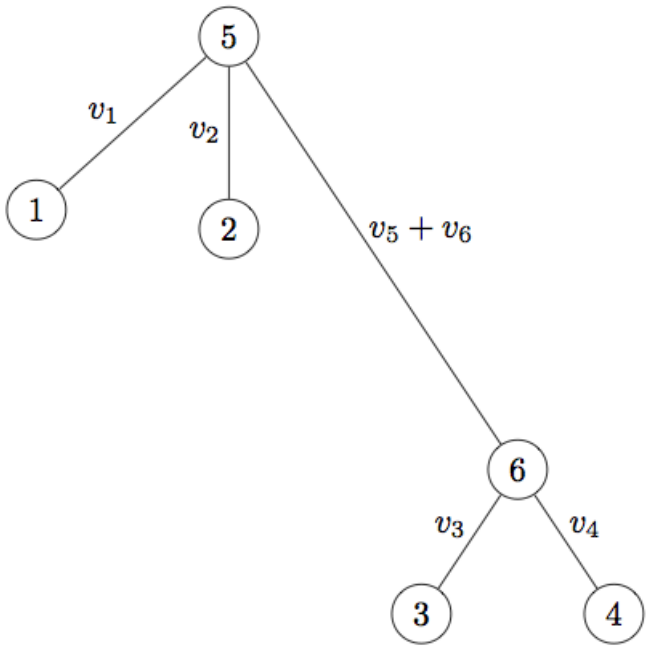


Figure 2.6: The tree in Figure 2.5 with the node 0 superimposed on node 5.

as the tree in Figure 2.6. In both cases we are calculating the likelihood of all trees of the family as shown in Figure 2.7. This is called the 'pulley principle' [14].

By the pulley principle, we obtain a computationally tractable method for obtaining the Maximum Likelihood Estimate (MLE) for the branch lengths. By varying each branch length v_i sequentially, and finding the optimal branch length for the unrooted tree, we find the tree with maximum likelihood. We can be certain of not entering a loop, as each iteration can only increase the likelihood.

Note that the above model of base substitution, where all substitutions are equally likely regardless of initial base type, is called Felsenstein's F81 model. For more complicated models than the F81 model of base substitution, a state aware $P_{ij}(t)$ can be found that truly depends on i. All time-reversible substitution models can be generalised to the Generalised time-reversible (GTR) model of substitution [41], so it remains only to discuss this model.

The GTR model has a symmetric instantaneous rate matrix form where each

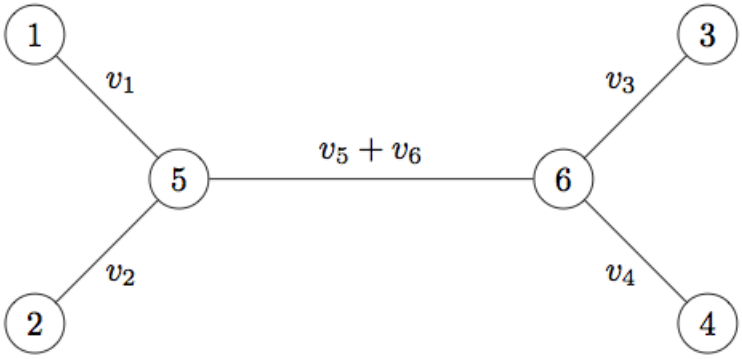


Figure 2.7: The unrooted tree from Figure 2.5 and Figure 2.6.

possible substitution rate $q_{ij} = q_{ji}$ can be distinct.

Now we have

$$P_{ij}(\Delta t) = P(t)\left(I + Q\Delta t\right), \tag{2.8}$$

where I is the identity matrix of appropriate dimension, and $[Q] = q_{ij}$ is the substitution rate matrix. A solution to equation (2.8) is

$$P(t) = e^{Qt},$$

and we substitute this solution for $P_{ij}(t)$ into equation (2.2). The remaining analysis then follows directly.

It should be noted that the above method will obtain the maximum likelihood tree and topology by using an exhaustive search through the possible topological tree space. For an unrooted bifurcating tree with $n \geq 3$ leaves, we have

$$U(n) = \frac{(2n-5)!}{(n-3)!(2^{n-3})}$$

possible tree topologies [15].

So, for a sample of size $n = 15$, we have $U(15) = 7.905854 \times 10^{12}$ possible trees to check. Clearly this is computationally intractable, and so heuristic methods for adding nodes in different orders, and hence searching tree topologies and comparing likelihoods to find an approximate maximum likelihood tree exist. For an exhaustive discussion on the topic, see [15].

When the most likely underlying genealogy has been generated for a set of sequences, 'known' coalescent event and inter-coalescent times are recovered. This genealogy we call $\mathbf{g}$, and it is the backbone of current processes used to recover estimates of the breeding population size changes over time.

In the following chapter we describe a common method for the recovery of population size estimates over time called Skyline Plots. Following this we introduce Approximated Bayesian Computation, a more recent tool for attempting to recover population size estimates over time.

Chapter 3

Population Estimation via the Skyline Plot

In Chapter 2 we described the Coalescent; a retrospective model of population genetics that employs the Wright-Fisher reproduction model backwards in discrete time. We then described the Coalescent Approximation, a continuous-time diffusion approximation of the Coalescent, which works well for large population sizes. However, to perform a coalescent analysis on a sample of sequences, we require a description of population sizes over time *a priori*. Since this is rarely known, we require a method of obtaining a sensible estimate of effective population sizes using only the sample of sequences.

As a first step we described a maximum likelihood method for obtaining the most likely genealogy of a given set of sequences without knowledge of the effective population size. The following 'Skyline Plot' methods describe the process by which we make estimates of population size dynamics from a genealogy.

Skyline Plot methods can be applied to any genealogy, but for the sake of the continuity of the description of current methods, we will be assuming the genealogy is the MLE genealogy $\mathbf{g}$. We will treat this genealogy as the true 'known' genealogy

of the sequences.

3.1 Classical Skyline Plots

Since we consider the population to be modelled in the framework of the Wright-Fisher reproduction method, we have that $N_e(x) = N(x)$. That is, the effective population is equal to the population itself.

We consider a set of n gene sequences sampled from a population at present. From the MLE methods in Section 2.2, the 'known' genealogy will have $n-1$ ordered nodes (coalescent events) labelled $I_2, I_3, \ldots, I_n$ with coalescent event times $t_2, t_3, \ldots, t_{n+1}$ and inter-coalescent event times $u_2, u_3, \ldots, u_n$, such that $u_i = t_i - t_{i+1}$ and $t_{n+1} = 0$ (see Figure 3.1).

From Section 2.1.2, we have that the rate of coalescence for k distinct lineages is

$$\frac{\binom{k}{2}}{N(t)} = \frac{k(k-1)}{2N_e(t)}.$$

Since the $n-1$ inter-coalescent event times are independent, we have that the probability the $(k-1)^{th}$ inter-coalescent event time since the MRCA equals u_k,

$$L(u_k \,|t_k\,) = \frac{k(k-1)}{2N_e(u_k + t_k)} \exp\left(- \int_{t_k}^{u_k + t_k} \frac{k(k-1)}{2N_e(t)} dt \right), \tag{3.1}$$

given that the interval begins at time t_k [25].

We will now derive estimates for population sizes $\boldsymbol{\Theta} = \{\theta_2, \ldots, \theta_n\}$ based on this likelihood function, during these inter-coalescent events for first isochronous (all sequences come from one sampling time), and then heterochronous samples (sampling times may vary for sequences). Finally we will modify these results to derive commonly used variants of the Skyline Plot.

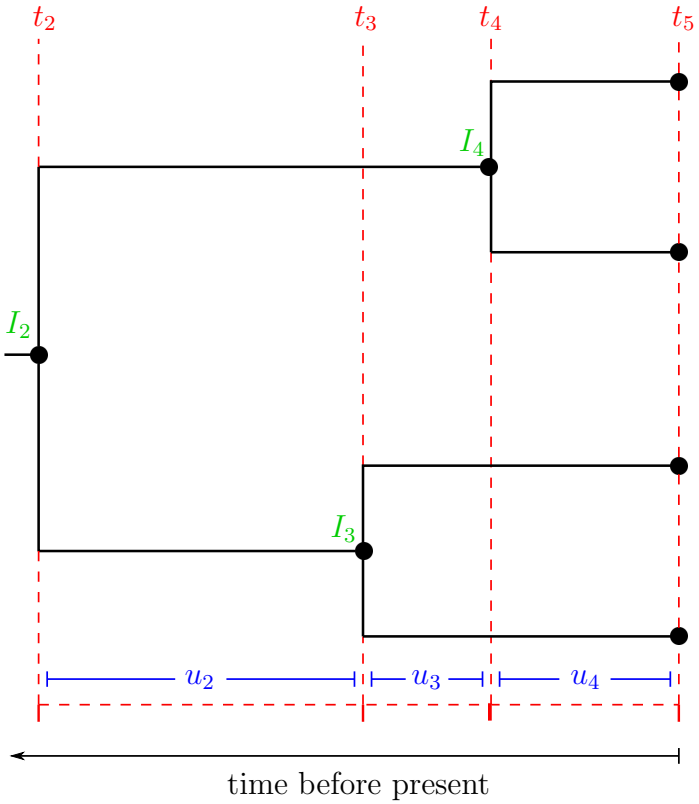


Figure 3.1: An example of a 'known' genealogy for $n = 4$ with isochronous samples.

3.1.1 Isochronous Generalised Skyline Plots

We assume that all samples were taken at the same time, and that the population size $N_e(t)$ can only change at a coalescent event. That is, we have that $N_e(t) = \theta_k$, where $\theta_k > 0$, $t_k < t < t_{k-1}$ and $k = 2, .., n$.

Now applying this new assumption to Equation (3.1) gives

$$\begin{aligned} L(\theta_k \,|\mathbf{g}\,) &= \frac{k(k-1)}{2\theta_k}\exp\left(-\int_{t_k}^{u_k+t_k}\frac{k(k-1)}{2\theta_k}dt\right) \\ &= \frac{k(k-1)}{2\theta_k}\exp\left(-\frac{k(k-1)}{2\theta_k}\int_{t_k}^{u_k+t_k}1dt\right) \\ &= \frac{k(k-1)}{2\theta_k}\exp\left(-\frac{u_k k(k-1)}{2\theta_k}\right). \end{aligned}$$

If we assume that the effective population sizes $\mathbf{\Theta} = \{\theta_2, \ldots, \theta_n\}$ are independent, then our joint likelihood for the genealogy $\mathbf{g}$, given the $n-1$ effective population sizes at event times of a sample of n individuals is,

$$L\left(\mathbf{g}\,|\mathbf{\Theta}\,\right) = \prod_{k=2}^{n}\frac{k(k-1)}{2\theta_k}\exp\left(-\frac{u_k k(k-1)}{2\theta_k}\right).$$

To obtain a maximum likelihood estimate $\hat{\theta}_k$ for θ_k, we take the log-likelihood

$$l(u_k\,|\theta_k\,) = \ln\left(k(k-1)\right) - \ln\left(\theta_k\right) - ln\left(2\right) - \frac{u_k k(k-1)}{2\theta_k}.$$

The maximum occurs when the log-likelihood function equals zero, hence

$$\frac{\partial l}{\partial \theta_k} = -\frac{1}{\theta_k} + \frac{u_k k(k-1)}{2\theta_k^2} = 0,$$

giving a maximum likelihood estimate for θ_k,

$$\hat{\theta}_k = \frac{u_k k(k-1)}{2} = \binom{k}{2}u_k,$$

since

$$\left.\frac{\partial^2 l}{\partial \theta_k^2}\right|_{\theta_k=\hat{\theta_k}} = \frac{-4}{(u_k k(k-1))^2} < 0,$$

for $k, u_k > 0$.

To obtain a sensible biological interpretation of $\hat{\theta}_k$, consider the following. If $U \sim U(0,1)$, then solving

$$U = \exp\left[-\int_{t_k}^{u_k+t_k}\frac{k(k-1)}{2N_e(t)}dt\right], \tag{3.2}$$

will generate a random variate from equation (3.1).

We can then rearrange equation (3.2) to obtain

$$\frac{u_k k(k-1)}{2} = \hat{\theta}_k = -\ln(U) H_k,$$

where

$$H_k = \left(\int_{t_k}^{t_k+u_k} \frac{1}{u_k N_e(t)} dt \right)^{-1}.$$

H_k has the meaningful interpretation of being the harmonic mean of the effective population size over the time interval $(t_k, t_k + u_k)$. Since $U \sim (0, 1)$, then $-\ln(U)$ takes any value on $(0, \infty)$, and hence can be thought of as a random scaling factor for the non-random value H_k. $\hat{\theta}_k$ can be thought of as our best estimate of H_k, the harmonic mean of N_e over the k^{th} interval.

It is worth noting that the harmonic mean of n variates is defined as

$$H = \left(\frac{1}{n} \sum_{i=1}^{n} \frac{1}{x_i} \right)^{-1}$$

and tends strongly toward the smallest samples in magnitude (when compared to the arithmetic mean). Hence, when discussing recovered effective population sizes over an interval it is worth keeping in mind that you are possibly giving greater weight to smaller population sizes during the interval.

Such estimates are called 'Classical Skyline Plots' (CSPs) and were introduced in 2000 by Pybus *et al.*. CSPs are a plot of the $\hat{\theta}_k$ over time and as such, are a piecewise non-parametric estimate of the demographic history [33]. Since CSPs are an estimate of $n - 1$ parameters from only n observations, they have large variances due to this over-fitting. Hence, CPSs often look unrealistic around times when inter-coalescent times are small, or when effective population sizes change dramatically at a coalescent event (see Figure 3.2).

The 'Generalised Skyline Plot' (GSP), developed by Strimmer *et al.*, restricts the number of different population sizes over intervals [39]. Intervals of length less than

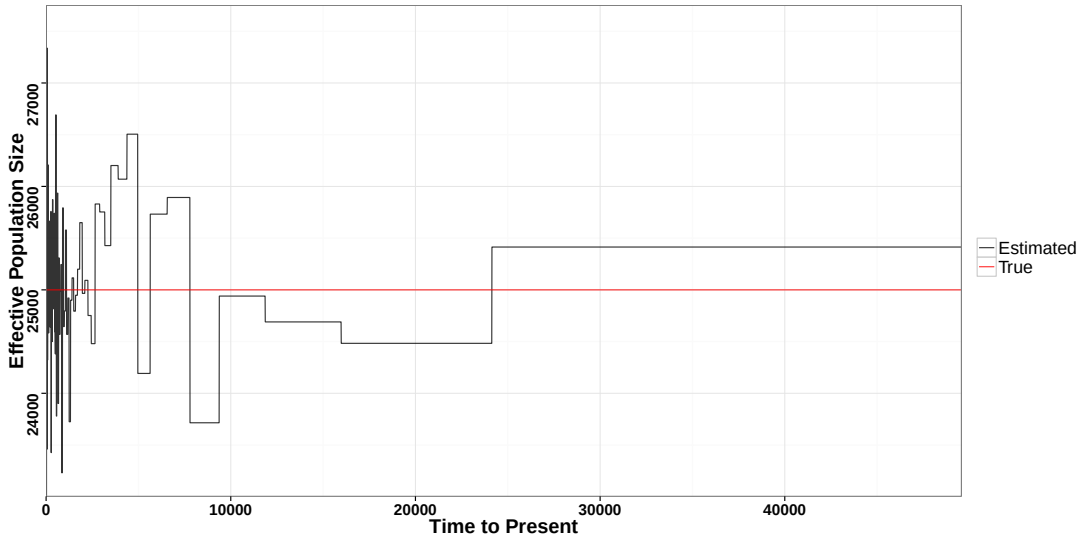


Figure 3.2: A realisation of a CSP for a constant population model and $n = 75$ and true effective population size 25000.

some predefined tolerance ϵ borrow their population size from neighbouring intervals. That is, the GSP require an ordered subset of group sizes, $A = \{a_2, \ldots, a_{m+1}\}$ (where $a_i \geq 1$ and $\sum_{\ell=1}^{m} a_\ell = n - 1$) that defines the number of coalescent events in each interval (see Figure 3.3).

The inter-coalescent times for each grouped interval are denoted $\mathbf{w} = \{w_2, \ldots, w_{m+1}\}$. We also define the function

$$h(i) := \begin{cases} 2 & \text{if } i \leq (a_2 + 1)\,, \\ j & \text{if } \left(\sum_{\ell=1}^{j-1} a_\ell\right) + 1 < i \leq \left(\sum_{\ell=1}^{j} a_\ell\right) + 1, \end{cases}$$

which is a map from the indices of $\mathbf{u}$ to the indices of $\mathbf{w}$.

We now have that the likelihood for our genealogy given the restricted $\boldsymbol{\Theta} = \{\theta_2, \ldots, \theta_n\}$ is

$$L\left(\mathbf{g}\,|\boldsymbol{\Theta}, \mathbf{A}\right) = \prod_{i=2}^{n} \frac{k_i(k_i-1)}{2\theta_{h(i)}} \exp\left(-\frac{u_i k_i(k_i-1)}{2\theta_{h(i)}}\right),$$

with associated log-likelihood function

$$l\left(\mathbf{g}\,|\boldsymbol{\Theta}, \mathbf{A}\right) = \sum_{i=2}^{n} \left\{ \ln\left(\frac{k_i(k_i-1)}{2}\right) - \ln(\theta_{h(i)}) - \frac{u_i k_i(k_i-1)}{2\theta_{h(i)}} \right\}$$

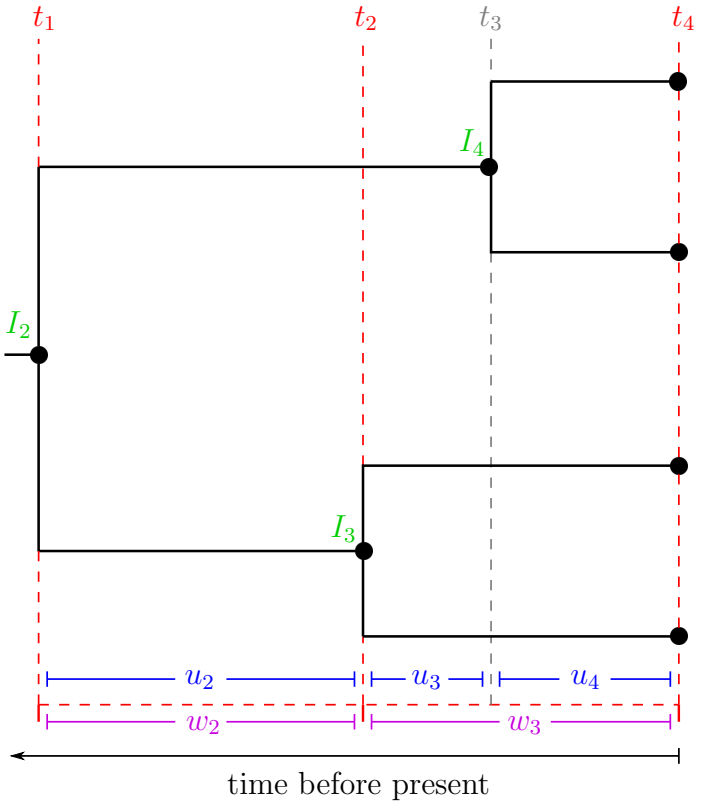


Figure 3.3: An example of a generalised genealogy with $n = 4$, $m = 2$ and $A = \{1, 2\}$ with isochronous samples.

Then for the set J_ℓ such that $h(j) = \ell \ \forall \ j \in J_\ell$,

$$\frac{\partial l}{\partial \theta_\ell} = \sum_{j \in J_\ell} \left[-\frac{1}{\theta_\ell} + \frac{u_j k_j (k_j - 1)}{2\theta_\ell} \right]$$

and by a similar argument as for the CSP, we find a maximum likelihood estimate for the ℓ^{th} interval for a GSP is

$$\hat{\theta}_\ell = \frac{\sum_{j \in J_\ell} u_j k_j (k_j - 1)}{2a_\ell}.$$

Note that $\hat{\theta}_\ell$ is just the weighted average of the MLE population sizes in the ℓ^{th} grouped interval since the MRCA.

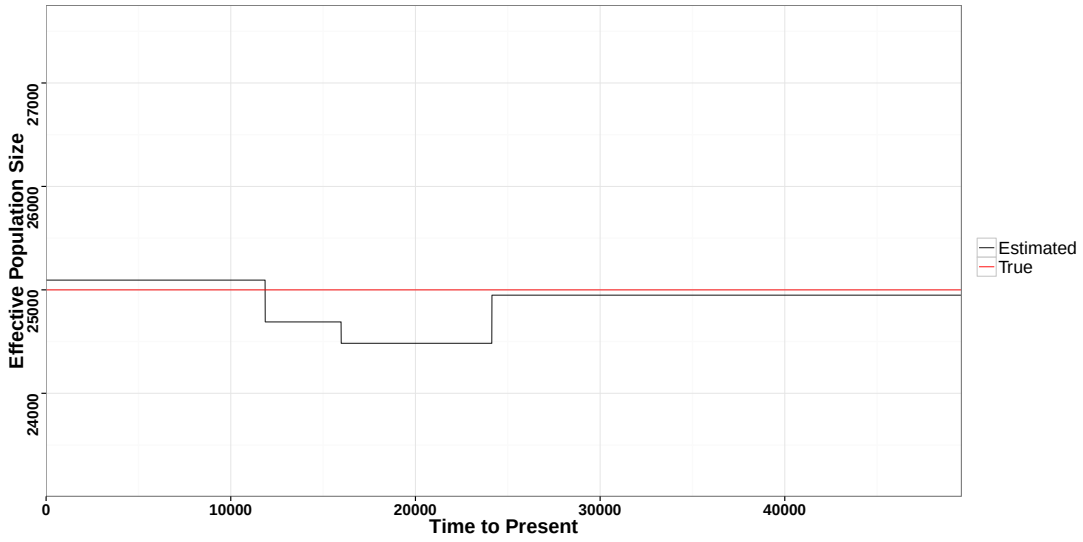


Figure 3.4: A GSP of the data from Figure 3.2 with $A = \{70, 1, 1, 1, 1\}$.

3.1.2 Heterochronous Generalised Skyline Plots

Heterochronous trees are much like the isochronous trees except that they have two types of intervals; coalescent intervals and sample intervals. Coalescent intervals end with a coalescent event, whereas sample intervals end with a sampled sequence (or sequences). See Figure 3.5. We present the work given in 1999 by Rodrigo et. al. [**?**].

For n sequences at s different sampling times, there will be $n+s-2$ intervals. The ordered inter-event times $\mathbf{u} = \{u_2, \ldots, u_{n+s-1}\}$ go from the most modern samples to the MRCA, and the indicator function

$$I_c(i) = \begin{cases} 1 & \text{if event } i \text{ is a coalescent event,} \\ 0 & \text{otherwise,} \end{cases}$$

is used to indicate the nature of event i. An interval ends in a coalescent event, or a sampling event, and we note that k_i can now increase or decrease depending on the nature of an event.

Again we take a set $\mathbf{w} = \{w_2, \ldots, w_{m+1}\}$, where the w_i are the grouped interval inter-event times, and we impose that all group intervals must end in a coalescent

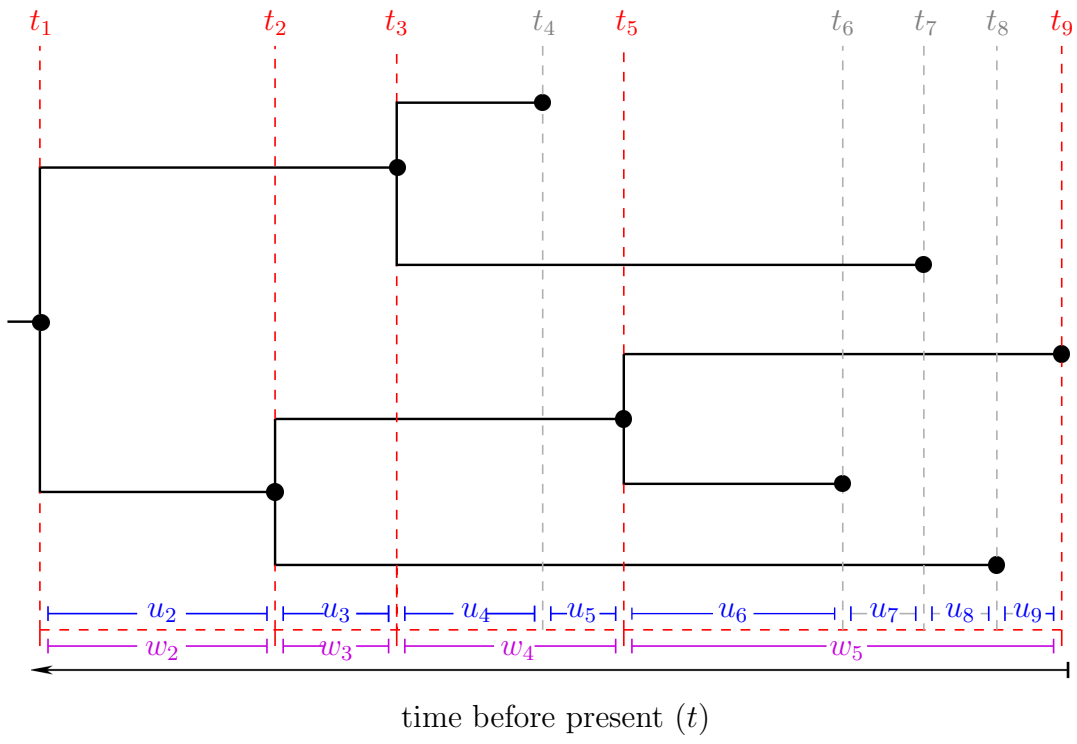


Figure 3.5: An example of a generalised genealogy for $n = 5$, $s = 5$, $m = 4$ and $A = \{1, 1, 1, 1\}$ for heterochronous samples.

event. Finally, we define the mapping function as

$$h(i) := \begin{cases} 2 & \text{if } \sum_{j=2}^{i} I_c(j) \leq a_2 + 1, \\ j & \text{if } \left(\sum_{\ell=2}^{j-1} a_\ell\right) + 1 < \sum_{j=2}^{i} I_c(j) \leq \left(\sum_{\ell=1}^{j} a_\ell\right) + 1. \end{cases}$$

We now have that for the grouped population sizes $\boldsymbol{\Theta} = \{\theta_2, \ldots, \theta_m\}$ that the effective population size does not change for a sample-ended interval.

The likelihood is now,

$$L\left(\mathbf{g} \,|\boldsymbol{\Theta}, \mathbf{A}\right) = \prod_{i=2}^{n+s-1} \left(\frac{k_i(k_i-1)}{2\theta_{h(i)}}\right)^{I_c(i)} \exp\left(-\frac{u_i k_i(k_i-1)}{2\theta_{h(i)}}\right),$$

with maximum likelihood estimate for the ℓ^{th} interval for a GSP,

$$\hat{\theta}_\ell = \frac{\sum_{j \in J_\ell} u_j k_j(k_j-1)}{2\sum_{j \in J} I_c(j)}.$$

This is again a weighted average of the MLE population sizes in the ℓ^{th} grouped interval since the MRCA, except with no weight given to sampling intervals.

Note that the likelihood for the case of heterochronous data can be generalised to the likelihood of isochronous data by setting $s = 1$ (only modern samples were taken), and $I_c(i) = 1 \ \forall \ i \in \{1, \ldots, n-1\}$.

Using heterochronous data allows estimates of branch lengths to be measured in units of time instead of generations. This allows the mutation rate μ to be estimated directly. We also receive extra information about population sizes from the sampling events, although this increase in information is not dramatic as there will still only be $n-1$ distinct coalescent events.

3.1.3 The Bayesian Skyride Plot and Further Modifications

Now that we have a method that avoids over-fitting the skyline plot, we wish to smooth out estimates for $\boldsymbol{\Theta}$ using a belief that the effective population sizes will be autocorrelated through time.

Since the likelihood function for the heterochronous case generalises to the likelihood function for the isochronous case, we can assume that data comes in the form of the inter-coalescent times $\mathbf{w} = \{w_1, \ldots, w_{n-1}\}$.

We consider the joint likelihood

$$P(\mathbf{w} \,|\boldsymbol{\Theta}) = \prod_{k=1}^{n-1} P(w_k \,|\theta_k),$$

where $\boldsymbol{\Theta} = \{\theta_1, \ldots, \theta_{n-1}\}$ are the effective population sizes.

Since we wish to make the assumption that effective population size changes are continuous through time, we introduce an exponential prior for θ_j,

$$\theta_j \sim \text{Exp}(\theta_{j-1}), \quad 2 \leq j \leq n-1,$$

and we introduce a scale-invariant prior for θ_1 since we believe that our prior is invariant to changes in time scale [10],

$$f_{\theta_1}(\theta_1) \propto \frac{1}{\theta_1}.$$

Markov Chain Monte Carlo Implementation

It is important to note that we do not know our genealogy $\mathbf{g}$, since we do not actually observe our genealogy. We actually observe some sequence data $\mathbf{D}$, sampled from some population.

Since sequence data is the result of some mutational process $\mathbf{Q}$ say, we instead make inferences about $P(\mathbf{D}\,|\mathbf{g}, \mathbf{Q}\,)$, and Bayes' Theorem indicates that

$$P(\mathbf{g}, \mathbf{Q}, \mathbf{\Theta}\,|\mathbf{D}\,) \propto P(\mathbf{D}\,|\mathbf{g}, \mathbf{Q}, \mathbf{\Theta}\,)P(\mathbf{g}, \mathbf{Q}, \mathbf{\Theta}). \tag{3.3}$$

Consider now how we employ the coalescent model. From a sample of sequences we go backwards in time, coalescing lineages one by one at coalescent event times, and adding lineages at sampling event times until we find the MRCA. This produces a genealogy $\mathbf{g}$ with inter-event times dependent on $\mathbf{\Theta}$, the set of (transformed) effective population size harmonic means over the event intervals.

Now, from the MRCA in our reconstructed genealogy $\mathbf{g}$ we apply $\mathbf{Q}$ forwards in time to produce our data $\mathbf{D}$. Thinking about our data $\mathbf{D}$ in this way has three important consequences.

First, our genealogy $\mathbf{g}$ together with our vector of (transformed) effective population size harmonic means $\mathbf{\Theta}$ are independent of the process of mutation. That is,

$$P(\mathbf{g}, \mathbf{Q}, \mathbf{\Theta}) = P(\mathbf{Q})P(\mathbf{g}, \mathbf{\Theta}). \tag{3.4}$$

Second, once we have $\mathbf{g}$ and $\mathbf{Q}$, the data $\mathbf{D}$ is independent of $\mathbf{\Theta}$ since the process

of mutation on any given lineage is unaffected by the effective population size at any time. That is,

$$P(\mathbf{D}\,|\mathbf{g},\mathbf{Q},\mathbf{\Theta}) = P(\mathbf{D}\,|\mathbf{g},\mathbf{Q}). \tag{3.5}$$

Last, by applying the results in equations (3.4) and (3.5) to equation (3.3), we get

$$P(\mathbf{g},\mathbf{Q},\mathbf{\Theta}\,|D) \propto P(D\,|\mathbf{g},\mathbf{Q})P(\mathbf{Q})P(\mathbf{g}\,|\mathbf{\Theta})P(\mathbf{\Theta}).$$

By observing that

$$P(\mathbf{g}\,|\mathbf{\Theta}) \propto P(\mathbf{w}\,|\mathbf{\Theta}),$$

we obtain

$$P(\mathbf{g},\mathbf{Q},\mathbf{\Theta}\,|D) \propto P(D\,|\mathbf{g},\mathbf{Q})P(\mathbf{Q})P(\mathbf{w}\,|\mathbf{\Theta})P(\mathbf{\Theta}).$$

Using a MCMC method, one will obtain a (predetermined) number of r samples from the posterior distribution of $\mathbf{g}$, $\mathbf{Q}$ and $\mathbf{\Theta}$, and we use these to make inferences about the population dynamics and genealogies simultaneously.

From MCMC to The Plotted Data

From the r sampled posterior genealogies, BEAST (a commonly used software package for performing coalescent analyses) has a sample of r estimates of T_{MRCA} [9]. From these, BEAST calculates the lower limit for the 95% probability interval for the T_{MRCA}, which we call T_{lower}. BEAST now calculates the 'plot times' at which it will estimate population sizes,

$$\mathbf{T}_{BEAST} = \{0, \frac{T_{lower}}{99}, 2\times\frac{T_{lower}}{99}, 3\times\frac{T_{lower}}{99}, \cdots, 98\times\frac{T_{lower}}{99}, T_{lower}\},$$

a vector of length one hundred from zero to T_{lower}.

From each genealogy BEAST samples, a piecewise estimate of the (constant) harmonic mean population size between consecutive coalescent events is obtained (see Figure 3.6 (a)). These estimates of the population sizes over time are a function of the genealogy, and the coalescent event times (interval end points) are fixed

for each genealogy, with no consideration given to the variability for these event times.

At each plot time $t \in \mathbf{T}_{BEAST}$, BEAST has a sample of effective population size estimates (one from each posterior genealogy). BEAST then plots the median and the upper and lower 95% probability interval limits for the effective population size estimates at that time. For an example of how this is done, see Figure 3.6.

Units for Recovered Posterior Parameters

Note that the units on the axes of a recovered Skyline Plot are of importance, and change depending on the data and parameters. For isochronous data, the substitution rate determines the units of time as well as the units of the recovered population size estimates. If the substitution rate is set to 1.0 (as is done by default in many software packages such as BEAST), then time will be measured in substitutions per site. Similarly the effective population size parameter estimate will be an estimate of $N_e\mu$, which is half of the standard population genetic parameter $\theta = 2N_e\mu$.

If the substitution rate is in mutations per site per year, then time will be measured in years, and the effective population size parameter estimate is now measured in units of $N_e\tau$, where τ is the generation length in years.

Finally, if the substitution rate is given in mutations per site per generation, then time will be measured in units of generations, and the effective population size parameter estimate is now measured in natural units. That is we are estimating N_e directly.

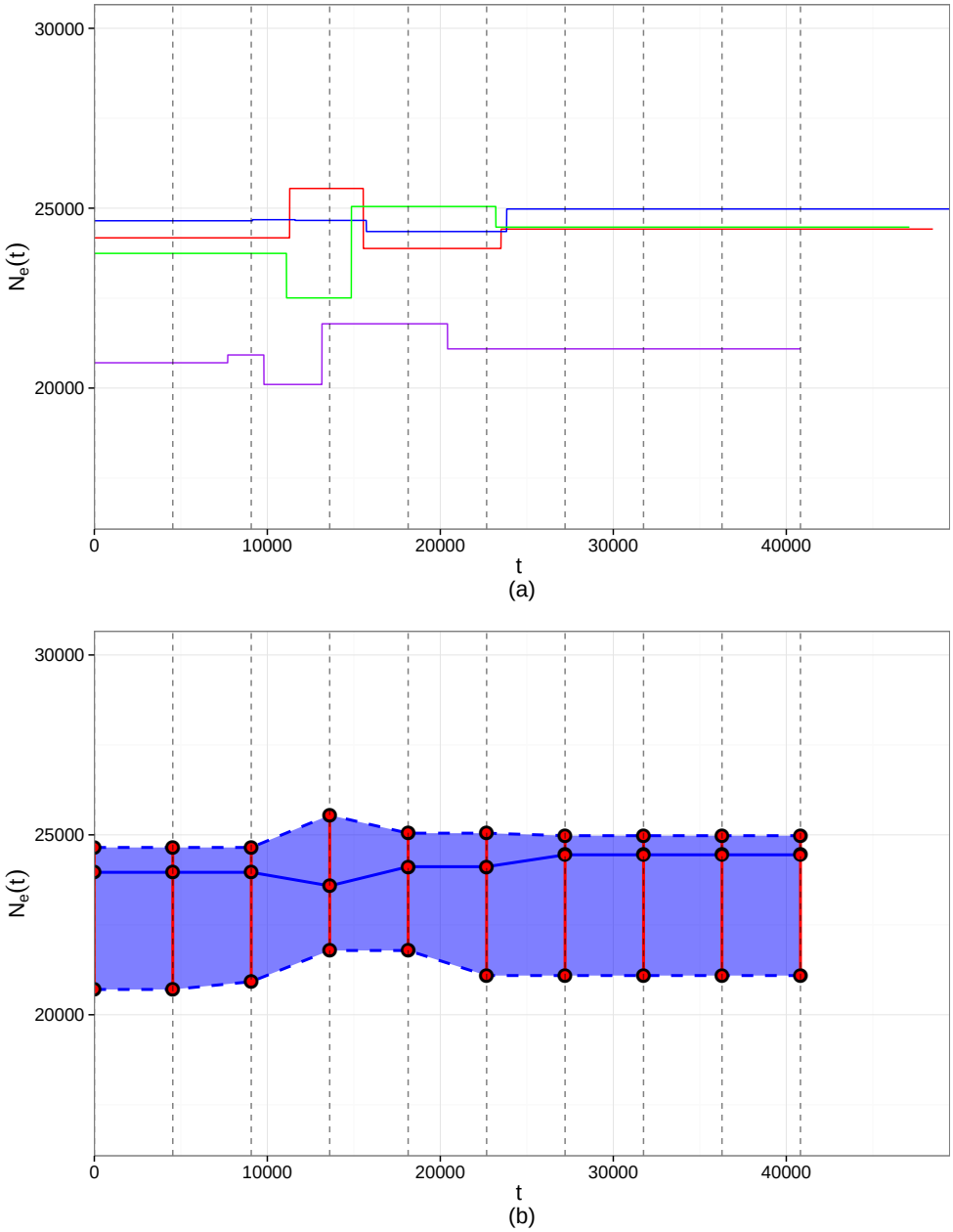


Figure 3.6: An example of how BEAST produces a single BSP for several posterior genealogies. In (a) we have four BSPs recovered from four sampled genealogies, and the vertical dashed lines are the plot times (we use 10 plot times, BEAST uses 100). In (b) we take the four recovered population size estimates at each plot time (the 10 dashed lines), and plot the median effective population size (the solid blue line), and the 95% probability intervals (the dashed blue lines). The blue shaded area is the 95% probability region (note, the 95% probability interval here is the range).

3.1.4 Model Selection within the Skyline Plot and the "Known" Tree

For the purpose of coalescent analyses, we must supply our model with a known population size model over time. To obtain this population model, we employ Felsenstein's method for the most likely underlying tree, and treat the interval times recovered from this process as true. Using these interval times, and one of the Skyline Plot methods, we acquire estimates of the (harmonic) mean population sizes over the recovered interval times.

If our population has been estimated to: be constant over time, have undergone an exponential decay/increase over time, or perhaps have suffered a bottle-neck event, then this should be straight forward to identify in the Skyline Plot. Estimates of the parameters can also be calculated from the vector of population sizes using standard MLE methods. We can then give this 'most likely' population model to our coalescent analysis (or present these results if this was the purpose of our study).

Note that a Skyline Plot method does not visually suggest population structure other than that of a single breeding population. Section 2.1.3 describes the migration model in which several sub-populations interbreed at some rate, and so it is possible to perform a coalescent analysis based on this population structure. It is not immediately obvious how one should go about selecting sub-population sizes, or migration rates, and this is just one example of a population structure that can not be visualised through any reconstructed skyline plot.

In Figure 3.7, a known constant population of 3000 individuals was simulated (using the software TreeSimJ [31]) and subdivided into two equal sized breeding groups, with a 'migration rate' of 0.01 for both groups. Four samples were analysed separately. Sample 1 contained just 25 modern sequences. Sample 2 contained 25 modern sequences, and 5 sequences from 1000 generations before present. Sample

3 contained 25 modern sequences, and 5 sequences from 1000, and 5 sequences from 3000 generations before present. Sample 4 contained 25 modern sequences, and 5 sequences from 1000, 5 sequences from 3000 and 5 sequences from 8000 generations before present.

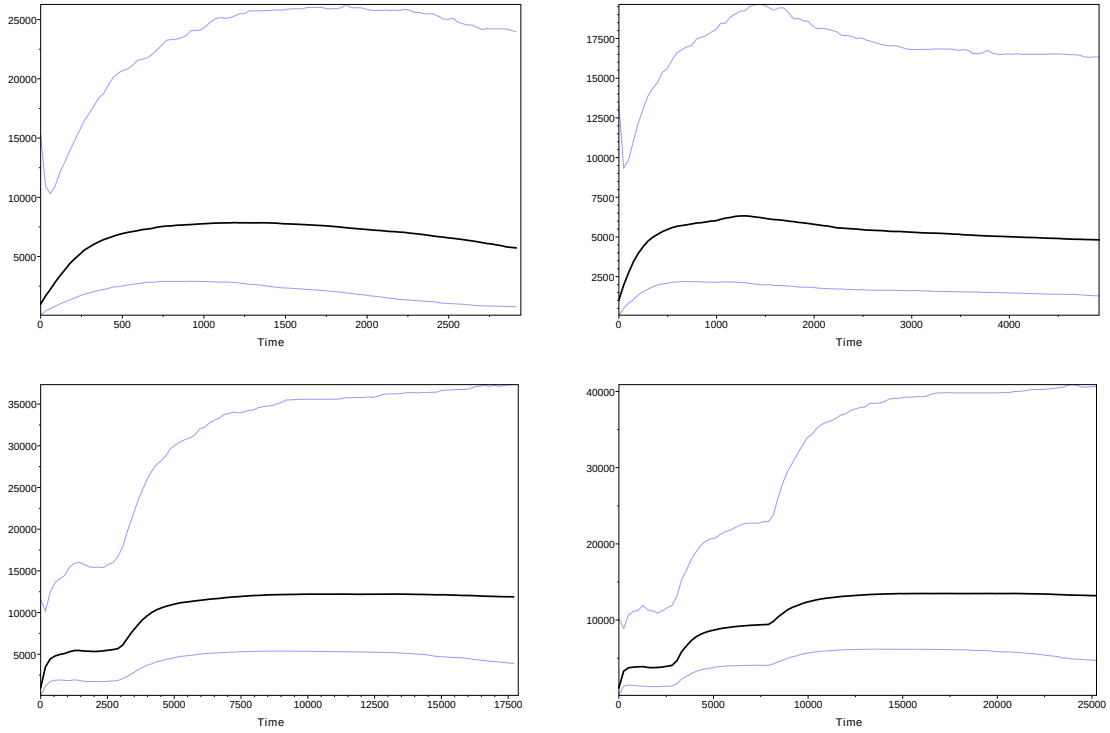


Figure 3.7: Reproduced Bayesian Skyline Plots for a two migrating sub-populations.

Immediately it can be observed that the population dynamics are being driven by the introduction of samples as for each successive plot, a 'step' is introduced each time sequences are introduced at the relevant sampling time (see Figure 3.7). However, it is not unreasonable to suggest that these are possible population dynamics over time for a single breeding population.

When selecting a population model, there is no way of telling how well your reconstruction of the population dynamics have fit. That is, you select a model visually, then fit your model parameters via some maximum likelihood method, but you

are given no indication of how well that structure has fit. In experiments where the entire point of the analysis was simply the reconstruction of the population size dynamics themselves, often some form of 95% confidence intervals are fit to the estimated effective population size, yet no indication is given for the variation in estimates of the event times themselves. For these reasons, we believe there is no real method for performing model selection on our data.

Felsenstein's method has one final assumption which may not seem plausible. We assume that there is only one most likely underlying tree that gave rise to our observed sequences, and that it is far more likely than any of the other possible underlying genealogical trees. Given the magnitude of the topological tree space, this can not be verified, and heuristic methods for obtaining our tree are necessary. However, this heuristically retrieved local maximum likelihood tree is treated as the global maximum likelihood tree, and hence the recovered event times and branch lengths are treated as known and true.

A similar argument exists for Bayesian Skyline Plot methods which sample from a 'tree space' for each MCMC step. BEAST employs 'operators' to make minor adjustments to the tree at each step. For example, the order of coalescence of samples may be rearranged locally. This subtle tree altering does not depart from the argument that the tree retrieved is a local maximum likelihood tree and also introduces another source of autocorrelation in the process. It does however allow for the labelled tree topology (the tree with labelled leaves) to be variable throughout the process.

Methods exist for describing how well your MLE tree fits your sequences [30], and so model checking can be performed on your underlying tree. However, methods for examining your underlying tree fit will not address questions of goodness of fit for population structure. We will suggest a likelihood free method using Approximate Bayesian Computation that does not suffer from the assumption of a most likely underlying tree, but will retrieve model parameters while employing a data driven

goodness of fit method for population structure (model) selection.

3.2 Approximate Bayesian Computation

Approximate Bayesian Computation (ABC) is a likelihood free method for obtaining samples from the posterior likelihood of some parameter set $\mathbf{\Theta} \in \mathbf{\Omega}$, given some observed data set $\mathbf{x}$. ABC is useful for problems where the likelihood function is difficult, or impossible to obtain, but we are able to efficiently obtain simulations of the process.

ABC arose in population genetics due to the intractability of the likelihood functions. Under the coalescent model, simulated sequences from specified populations models can be easily produced (for examples, see Section 2.1.2), and hence an ABC scheme for parameter estimation, given a population model, seems reasonable.

Here we introduce ABC, and provide a brief background on its development. Initially we describe the Acceptance-Rejection algorithm, one of the earliest ABC algorithms. We then introduce sufficient summary statistics and their use in ABC, and implement the use of sufficient summary statistics in a modified version of the Acceptance-Rejection algorithm. This will lead to a discussion on an automated method for obtaining the 'best' summary statistics when sufficient statistics are not available.

3.2.1 The Acceptance-Rejection Algorithm

Consider the posterior distribution for the parameters $\mathbf{\Theta}$,

$$p(\mathbf{\Theta}\,|\mathbf{x}) = \frac{f(\mathbf{x}\,|\mathbf{\Theta})r(\mathbf{\Theta})}{p(\mathbf{x})}, \tag{3.6}$$

where $r(\mathbf{\Theta})$ is the prior distribution for the parameters $\mathbf{\Theta}$, $f(\mathbf{x}\,|\mathbf{\Theta})$ is the likelihood of observing the data $\mathbf{x}$ given the parameters $\mathbf{\Theta}$, and $p(\mathbf{x})$ is the probability of observing the data $\mathbf{x}$.

In the simplest form of ABC, the acceptance-rejection algorithm for obtaining k posterior samples is given in Algorithm 1 An exhaustive discussion on the history of the acceptance-rejection algorithm, and the variations that followed, can be found in [4].

Algorithm 1: The acceptance-rejection algorithm.

```
Set i = 0;
while i < k do
    Sample Θ* from r(Θ);
    Simulate a realisation of the process x* from f(x|Θ*);
    if x* = x then
        accept Θ*;
        i = i + 1;
    end
end
```

The acceptance-rejection algorithm generates k samples from the posterior distribution $p(\boldsymbol{\Theta}\,|\mathbf{x})$ [37]. Note though that this algorithm relies on producing an exact copy of the observed data. In the case of sequence data, when we produce a coalescent based simulation, we take an arbitrary sequence as the sequence for our MRCA, and then apply mutations forwards in time given the branch lengths. For each branch we apply this process independently, and we take 'samples' as needed at different times. The probability of applying exactly the same mutations on any two realisations of the process is so small that the acceptance-rejection algorithm would take an extremely long time.

Similarly, we do not have the MRCA sequence from the real data set to which we can apply mutations. So we can not start our process from the same inherited sequence as the observed data has. This makes it impossible to compare raw sequences. We therefore need some other way of comparing simulated sequence

sets to the observed data, and to achieve this we employ summary statistics.

3.2.2 ABC using Sufficient Summary Statistics

Consider the data $\mathbf{x}$ whose distribution is conditional upon the parameters $\boldsymbol{\Theta}$. A summary statistic $S(\mathbf{x})$ is some function of the data $\mathbf{x}$ and is of the form

$$S : \mathbb{R}^d \to \mathbb{R}^w$$

where $d >> w$, and hence we reduce the dimensionality of the representation of our data. $S(\mathbf{x})$ is called a sufficient statistic if the conditional likelihood of the data $\mathbf{x}$, given $S(\mathbf{x})$, does not depend on the parameters $\boldsymbol{\Theta}$. In Fisher-Neyman factorised form we have

$$f(\mathbf{x}\,|\boldsymbol{\Theta}) = h(\mathbf{x}) g_{\boldsymbol{\Theta}}(S(\mathbf{x})\,|\boldsymbol{\Theta})$$

where $h(\mathbf{x})$ does not depend on $\boldsymbol{\Theta}$, and $g_{\boldsymbol{\Theta}}(S(\mathbf{x})\,|\boldsymbol{\Theta})$ depends on $\mathbf{x}$ only through $S(\mathbf{x})$, and is the likelihood of the sufficient statistic. Note then that $g_{\boldsymbol{\Theta}}(\cdot)$ carries all of the information about $\boldsymbol{\Theta}$.

Rearranging equation (3.6) we have that

$$\begin{aligned}
p(\boldsymbol{\Theta}\,|\mathbf{x}) &= \frac{f(\mathbf{x}\,|\boldsymbol{\Theta})r(\boldsymbol{\Theta})}{p(\mathbf{x})} \\
&= \frac{h(\mathbf{x}) g_{\boldsymbol{\Theta}}(S(\mathbf{x})\,|\boldsymbol{\Theta})r(\boldsymbol{\Theta})}{\int_\Omega h(\mathbf{x}) g_{\boldsymbol{\Theta}}(S(\mathbf{x})\,|\boldsymbol{\Theta})r(\boldsymbol{\Theta})d\theta} \\
&= \frac{g_{\boldsymbol{\Theta}}(S(\mathbf{x})\,|\boldsymbol{\Theta})r(\boldsymbol{\Theta})}{\int_\Omega g_{\boldsymbol{\Theta}}(S(\mathbf{x})\,|\boldsymbol{\Theta})r(\boldsymbol{\Theta})d\theta} \\
&= \frac{g_{\boldsymbol{\Theta}}(S(\mathbf{x})\,|\boldsymbol{\Theta})r(\boldsymbol{\Theta})}{p\left(S(\mathbf{x})\right)} \\
&= p\left(\boldsymbol{\Theta}\,|S(\mathbf{x})\right).
\end{aligned}$$

So our posterior distribution for the parameters $\boldsymbol{\Theta}$ given the data $\mathbf{x}$ is identical to the posterior distribution for the parameters $\boldsymbol{\Theta}$ given the sufficient statistic $S(\mathbf{x})$.

Now we decide upon some distance function for the two observed summary statistics $\rho(\cdot, \cdot)$ with corresponding 'tolerance' ϵ, and modify the Acceptance-Rejection algorithm such that we have the ABC algorithm given in Algorithm 2.

Algorithm 2: The modified acceptance-rejection algorithm.

```
Set i = 0;
while i < k do
    Sample Θ* from r(Θ);
    Simulate a realisation of the process x* from f(x|Θ*);
    if ρ(S(x*), S(x)) < ε then
        accept Θ*;
        i = i + 1;
    end
end
```

The output from this ABC algorithm, called the ABC rejection sampler, is a sample of parameters from $p(\boldsymbol{\Theta} \,|\rho\,(S(\mathbf{x}), S(\mathbf{x}^*)) < \epsilon)$. If ϵ is sufficiently small, then $p(\boldsymbol{\Theta} \,|\rho\,(S(\mathbf{x}), S(\mathbf{x}^*)) < \epsilon)$ will be a good approximation of $p(\boldsymbol{\Theta} \,|\mathbf{x})$ [43].

If the prior distribution for the parameters is significantly different from the posterior distribution, the rate of acceptance will be very low and the algorithm will take extremely long to complete. To avoid this problem, an algorithm based on Markov chain Monte Carlo (MCMC) was introduced [24]. The ABC MCMC algorithm is defined in Algorithm 3

The stationary distribution of this algorithm is a Markov chain with stationary distribution $p(\boldsymbol{\Theta} \,|\rho\,(S(\mathbf{x}), S(\mathbf{x}^*)) < \epsilon)$ [24]. Note that this algorithm can suffer long run times due to the correlated nature of the sampling procedure and potentially low acceptance probabilities, and depending on the proposal distribution, can spend long periods in low probability regions [43].

Unfortunately this discussion on good estimates for posterior distributions hinges

Algorithm 3: The ABC MCMC algorithm.

Set $i = 0$ and initialise $\boldsymbol{\Theta}_i$;
while $i < k$ **do**
  Propose $\boldsymbol{\Theta}^*$ according to a proposal distribution $q(\boldsymbol{\Theta} \,|\boldsymbol{\Theta}_i)$;
  Simulate a realisation of the process $\mathbf{x}^*$ from $f(\mathbf{x} \,|\boldsymbol{\Theta}^*)$;
  Let $\alpha = \min\left(1, \frac{r(\boldsymbol{\Theta}^*)q(\boldsymbol{\Theta}_i|\boldsymbol{\Theta}^*)}{r(\boldsymbol{\Theta}_i)q(\boldsymbol{\Theta}^*|\boldsymbol{\Theta}_i)}\right)$;
  if $\rho\left(S(\mathbf{x}^*), S(\mathbf{x})\right) < \epsilon$ **then**
    $\boldsymbol{\Theta}_{i+1} = \boldsymbol{\Theta}^*$;
  else
    Sample $u \sim U(0, 1)$;
    if $u \leq \alpha$ **then**
      $\boldsymbol{\Theta}_{i+1} = \boldsymbol{\Theta}^*$;
    else
      $\boldsymbol{\Theta}_{i+1} = \boldsymbol{\Theta}_i$
    end
  end $i = i + 1$;
end

upon having sufficient summary statistics, and yet we have no such sufficient statistics for sequence data. In the next chapter we discuss a semi-automatic method for constructing summary statistics as given by Fearnhead *et al.* [13].

Chapter 4

Semi-Automatic Approximate Bayesian Computation

In Chapter 3 we described the final steps of Skyline Plot based methods for inferring population size dynamics from a sample of DNA sequences. In describing these methods, we highlighted two of the key problems with these techniques.

First, these models only allow for single breeding groups, but give no indication of how well a population model under this assumption fits the data. That is, we cannot identify any type of migration dynamics for separated sub-populations, yet we can specify this model when performing a follow-up coalescent analysis (in theory). Clearly then, one cannot implement a migration model without prior knowledge.

Second, one can question the validity of the 'local' MLE tree given that it is obtained via heuristic methods. This 'local' MLE tree is employed as the 'global' MLE tree in classical analyses, and in the case of the Bayesian Skyline Plot, the heuristically obtained local MLE tree is treated the same way.

We then introduced Approximate Bayesian Computation (ABC), and the use of sufficient summary statistics in ABC. We claim that we can use these ABC meth-

ods to circumvent the key problems with Skyline Plot methods. Since we do not have sufficient summary statistics for our DNA sequences, we employ the method introduced by Fearnhead *et al.* [13].

We begin this Chapter by describing this method, and the summary statistics common in DNA analysis. We then perform three analyses on three separate data sets generated under a Constant, Exponential and Migration population model, and report our results.

4.1 Common Summary Statistics for DNA

Several of the following summary statistics described are constant multiples of another summary statistic. For completeness we present all summary statistics commonly found in the literature for statistical DNA analysis, and where applicable, mention these relationships.

Consider a set of DNA sequences from two unique geographical regions within a larger area (similarly the DNA sequences might be separated by time). We begin by describing 'single sample summary statistics' which quantify some characteristic of the DNA sequences from each region. For example, we might obtain two values which represent the genetic diversity in each region.

We then describe 'multiple sample summary statistics' which quantify a comparison of the two regions. For example, we might obtain a single value which represents the number of alleles unique to the first region.

4.1.1 Single Sample Summary Statistics

The following summary statistics are calculated on a single sample of DNA sequences. Commonly these characteristics are either some measure of genetic diversity, or selection neutrality. In Section 4.1.2 we present summary statistics that

are used to compare two or more separate samples of sequences.

In Example 1 (see Figure 4.1) we have a single sample, which we call 'sample v', with $n = 4$ sequences of length 8 (that are written from left to right). The two allelic forms are coloured blue and green, and segregating sites are coloured red. We use Example 1 to demonstrate the calculation of each of the following single sample summary statistics.

Note that a subscript for any single sample summary statistic of the form v is calculated on the sample of sequences v, and that a subscript of the form $\{v, w\}$ is calculated on the pooled sample obtained by combining the sequences in v and w into one sample.

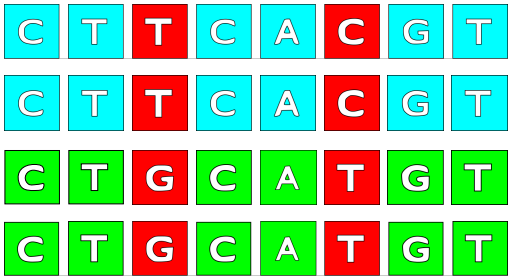


Figure 4.1: Example 1.

Haplotypes (h_v): h_v is defined as the number of unique sequences in the sample v, of size n_v sequences [27]. The number of haplotypes can be thought of as the number of distinct alleles in a sample. For Example 1 we find $h_v = 2$. That is, there are two haplotypes (one type in green, one type in blue).

Segregating Sites (S_v): The number of sequence elements in the sample sequences that are not homogeneous for all sequences in the sample v. Also called the number of single-nucleotide polymorphisms (SNPs) [47]. For Example 1 we find $S_v = 2$, the sequence positions coloured red.

Pairwise Differences ($\hat{k}_v$): The mean number of segregating sites for every

pair of sequences in the sample v [28]. A measure of nucleotide diversity, it is calculated as:

$$\hat{k}_v = \frac{\sum_{i<j} S_{\{i,j\}}}{\binom{n_v}{2}}$$

where $S_{\{i,j\}}$ is the number of segregating sites between sequences i and j, and n_v is the number of sequences in our sample. For Example 1 we find:

$$\begin{aligned}\hat{k}_v &= \frac{0+2+2+2+2+0}{\binom{4}{2}} \\ &= \frac{4}{3}.\end{aligned}$$

Haplotype Diversity (H_v): A measure of the diversity of unique haplotypes in a population [29]. Calculated for the sample v as:

$$H_v = \left(1 - \sum_{i=1}^{h_v} x_i^2\right),$$

where x_i is the relative frequency of haplotype i in the sample and h_v is the number of unique haplotypes in sample v. For Example 1 we find:

$$\begin{aligned}H_v &= \left(1 - \left[\left(\frac{2}{4}\right)^2 + \left(\frac{2}{4}\right)^2\right]\right) \\ &= \frac{1}{2}.\end{aligned}$$

Note that an alternative definition

$$H_v^* = \frac{n_v}{n_v - 1}\left(1 - \sum_{i=1}^{h_v} x_i^2\right)$$

exists, and is useful when comparing H_v^* for different sized samples. When comparing two equally sized samples, the two definitions contain the same amount of information and hence H_v is biased by a factor of $\frac{n_v-1}{n_v}$. We leave it in this form to conform with BayeSSC output (a popular software package that performs

coalescent simulations) [12]. (This bias can be ignored because, if we 'retain' H_v as significant, then the coefficient for this summary statistic in our linear model will be scaled accordingly, as we will see in the next section).

Nucleotide Diversity (π_v): A measure of the 'degree of polymorphism', or how often sequence element positions are not homogeneous across a population [28]. It is calculated for sample v as:

$$\pi_v = \frac{\sum_{i<j} S_{\{i,j\}}}{\ell \binom{n_v}{2}}$$

where $S_{\{i,j\}}$ is the number of segregating sites per site between sequences i and j, ℓ is the number of elements in our sequences, and n_v is the number of sequences in our sample. Note then that

$$\pi_v = \frac{1}{\ell}\hat{k}_v.$$

Hence for Example 1, we have that:

$$\begin{aligned}\pi_v &= \frac{1}{8}\hat{k}_v \\ &= \frac{1}{6}.\end{aligned}$$

Tajima's D (D_v): Tajima's D statistic aims to distinguish whether the region of DNA we are investigating has evolved under a neutral model, or under some other process, for example selection [40]. It compares two estimates of the expected number of SNPs by taking their difference and dividing them by their standard deviation, for the sample v. It is calculated as:

$$D_v = \frac{\hat{k}_v - \frac{S_v}{a_1}}{\sqrt{e_1 S + e_2 S_v(S_v - 1)}},$$

where S_v and $\hat{k}_v$ are the number of segregating sites and mean number of pairwise differences respectively for sample v, n_v is the total number of sequences in our sample and:

$e_1 = \frac{c_1}{a_1}$	$e_2 = \frac{c_2}{a_1^2}$
$c_1 = b_1 - \frac{1}{a_1}$	$c_2 = b_2 - \frac{n_v+2}{na_1} + \frac{a_2}{a_1^2}$
$b_1 = \frac{n_v+1}{3(n_v-1)}$	$b_2 = \frac{2(n_v^2+n_v+3)}{9n_v(n_v-1)}$
$a_1 = \sum_{i=1}^{n_v-1} \frac{1}{i}$	$a_2 = \sum_{i=1}^{n_v-1} \frac{1}{i^2}$

For Example 1 we calculate an observed $d_v = 1.893$.

4.1.2 Multiple Sample Summary Statistics

The following summary statistics are calculated on multiple grouped sequences, such as when we have modern and ancient samples. In Example 2 (see Figure 4.2) we have two samples, v and w, of size $n_v = 4$ and $n_w = 3$ sequences respectively. Group v is from Example 1 and is on the left. Group w is on the right. Note that segregating sites are only highlighted within groups. We use Example 2 to demonstrate the calculation of each of the following multiple sample summary statistics.

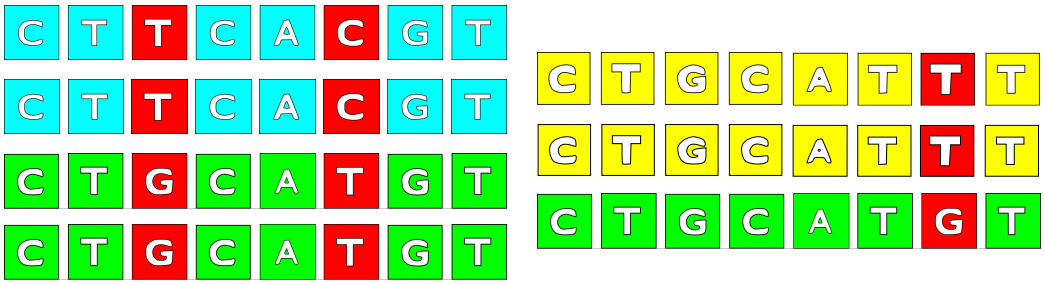


Figure 4.2: Example 2.

Private To v vs w ($p_{v,w}$):

The number of unique alleles that are found only in sample v when compared with sample w [38]. For Example 4.2 we have $p_{v,w} = 1$ (the blue sequences) and $p_{w,v} = 1$ (the yellow sequences). Note that the green sequences were in both samples, hence they were not private to either group.

Mean Haplotype Diversity ($\bar{H}_s$):

For m grouped samples, where H_i is the haplotype diversity of the i^{th} grouped sample, we define

$$\bar{H}_s = \frac{1}{m}\sum_{i=1}^{m} H_i$$

as the mean haplotype diversity [29].

For what follows we only calculate $\bar{H}_s$ for two specific samples at any given time, and denote this $\bar{H}_s^{(v,w)}$ for samples v and w, for example. For Example 2 we have $H_v = \frac{1}{2}$ and $H_w = \frac{4}{9}$, hence

$$\bar{H}_s^{(v,w)} = \frac{17}{36}.$$

Note that we have a biased estimate of the population mean haplotype diversity (by virtue of the factor of $\frac{n_v-1}{n_v}$ for group v). This bias is not equal for any groups v and w such that $n_v \neq n_w$, and hence our estimate of $\bar{H}_s$ is not equally weighted. We have found that correcting for this has little to no effect on the final outcome of simulations. For conformity with BayeSSC simulation software output, we do not correct for this bias.

Pooled Haplotype Diversity (H_T):

If we pool each of our n_P samples of sequences into one pooled sample, then for each of the m_P unique alleles in our pooled sample, we define

$$\bar{x}_i = \frac{1}{n_P}\sum_{j=1}^{n_P} x_{ij},$$

where x_{ij} is the proportion of sequences in group j that are of allelic type i, which is the mean allelic proportion of allele i across all groups.

Now we can define the pooled haplotype diversity[27] as

$$H_T = 1 - \sum_{i=1}^{m_P} \bar{x}_i^2.$$

Again, we denote $H_T^{(v,w)}$ as the pooled haplotype diversity for groups v and w. For Example 2 we find

$$H_T^{(v,w)} = \frac{47}{72}.$$

Gene Flow (F_{ST}): A measure of the mean amount of gene flow between different geographically separated sub-populations. We define

$$F_{ST} = \frac{H_T - \bar{H}_S}{H_T} = 1 - \frac{\bar{H}_S}{H_T},$$

as proposed by Hudson *et al.* [16].

In the context of DNA separated by time instead of geographical distance, the interpretation of F_{ST} becomes the mean amount of gene flow between generations over time.

If we define $F_{ST}^{(v,w)}$ as the gene flow between groups v and w, for Example 2 we find

$$F_{ST}^{(v,w)} = \frac{13}{47}.$$

In total there are nine 'categories' of summary statistic we consider (see Table 4.1). Note, we do not use the number of pairwise differences $\hat{k}_v$ as it is a scalar multiple of the nucleotide diversity π_v.

Name	Notation
Haplotypes	(h_v)
Segregating Sites	S_v
Haplotype Diversity	H_v
Nucleotide Diversity	π_v
Tajima's D	D_v
Private To	$p_{v,w}$
Mean Haplotype Diversity	$\bar{H}_s$
Pooled Haplotype Diversity	H_T
Gene Flow	F_{ST}

Table 4.1: Considered summary statistics.

4.2 Approximate Bayesian Computation with Constructed Summary Statistics

We follow the method suggested by Fearnhead *et al.*[13], and direct the reader to this paper for an exhaustive discussion of the topic.

Ideally we would like a sufficient summary statistic for our posterior mean. Since the posterior mean is a sufficient statistic for itself, we aim to define a linear combination of our observed summary statistics to best model our posterior mean. That is, we wish to build a linear model of our summary statistics that best predicts our parameters of interest.

In the following section we denote our data $\mathbf{x}$, our summary statistics $\mathbf{S} = (s_1, s_2, \ldots, s_T)$, which are some function of the data such that $h(\mathbf{x}) = \mathbf{S}$, and the parameters which define the distribution of our data $\mathbf{\Phi} = (\phi_1, \phi_2, \ldots, \phi_P)$.

Finally, recall from Section 2.1.1 that the first step in any coalescent simulation is to generate the required number of exponentially distributed inter-event times. We then rescale time according to some population dynamics model. The sum of the rescaled inter-event times give us an observed T_{MRCA}, and we use this to construct our posterior T_{MRCA} distribution.

4.2.1 Step 0: Obtain Observed Data Set

While this is an obvious step in any analysis, for the purpose of clarity when reporting results, we describe how we obtained our 'true data', and the parameters under which it was obtained. We call this observed data set 'ObsDat', and it is the sampled DNA about which we would like to make inferences.

4.2.2 Step 1: Constructing Approximately Sufficient Summary Statistics

We generate a large number Γ of simulations across a uniform grid of the support of the prior distribution $\mathbf{r}(\mathbf{\Phi})$ for our parameters of interest. For each simulated realisation we calculate a corresponding set of T summary statistics. We call this set of $\Gamma \times T$ summary statistics with known input parameters our training set, 'TrainDat'.

Now, for each $\phi_k \in \mathbf{\Phi}$, $k = \{1, 2, \ldots, P\}$, we perform a linear regression on the TrainDat such that

$$\hat{\phi}_k = \hat{\alpha}_0^{(k)} + \sum_{j=1}^{T} \hat{\alpha}_j^{(k)} s_j, \quad k = 1, \cdots, P.$$

We then employ a Box-Cox Transformation to $\hat{\phi}_k$ such that

$$\hat{\phi}_k^{(\lambda_k)} = \begin{cases} \frac{\hat{\phi}_k^{\lambda_k} - 1}{\lambda_k}, & \text{if } \lambda_k \neq 0, \\ \ln(\hat{\phi}_k), & \text{if } \lambda_k = 0 \end{cases},$$

such that λ_k maximises the profile-likelihood (where the profile-likelihood is the log-likelihood written in terms of just λ_k, and all other parameters are held constant). We do this to help satisfy the assumption of linearity in our model [6] (using, in the `R-Statistical Software Package`, the `boxcox()` function in the 'MASS' package [45]).

We then have that the predicted values for the parameters of interest are defined by the function $g(\cdot)$ such that,

$$\begin{aligned} \hat{\mathbf{\Phi}}' &= g(\mathbf{S}') \\ &= g(h(\mathbf{x}')) \\ &= \{(\hat{\phi}_1')^{\lambda_1}, \ldots, (\hat{\phi}_P')^{\lambda_P}\} \end{aligned}$$

is a set of predicted transformed posterior means for our P parameters of interest given some data set $\mathbf{x}'$ with associated summary statistics $h(\mathbf{x}') = \mathbf{S}'$.

Since it is unlikely that every one of our candidate summary statistics will be significant for predicting the posterior means, we employ a backwards step model selection algorithm (in the `R-Statistical Software Package`, using the 'stats package' `step()` function [34]). This method allows us to start with the (full) transformed model, and we sequentially remove predictor variables until a 'best' (with respect to AIC) model is found (see Appendix A.2 for a full description of the algorithm).

4.2.3 Step 2: ABC Using Constructed Summary Statistics

We define $\hat{\Phi}^{obs}$ as the predicted parameters of interest from our observed sample of sequences and let ϵ be the tolerance when comparing samples. We also define the 'distance' between two samples $\rho(\cdot, \cdot)$ as,

$$\rho(\mathbf{\Phi}^{(1)}, \mathbf{\Phi}^{(2)}) = \left[\sum_{k=1}^{P} \left(\hat{\phi}_k^{(1)} - \hat{\phi}_k^{(2)} \right)^2 \right]^{\frac{1}{2}},$$

the Euclidean distance between a vector of P predicted posterior means.

Next we select the number of posterior samples we wish to retain, N_{ABC}, and perform Algorithm 4.

We refer to the simulated data set employed for the purpose of ABC comparisons as the 'ABCDat'.

4.3 Results

We simulated three ObsDats using BayeSSC. Each ObsDat was simulated under the Jukes-Cantor model of DNA evolution, with sequence length 1000 bp, a

Algorithm 4: ABC using Constructed Summary Statistics

Input: ObsDat, TrainDat

Define $g(\cdot)$ using TrainDat;
Calculate $\hat{\Phi}^{obs} = g\left(h(\text{ObsDat})\right)$;
Set $i = 0$;
while $i < N_{ABC}$ **do**
    Sample $\mathbf{\Phi}^*$ from $\mathbf{r}(\mathbf{\Phi})$;
    Simulate a realisation $\mathbf{x}^*$ from $f(\mathbf{x}\,|\mathbf{\Phi}^*)$ and calculate $\hat{\Phi}^* = g(h(\mathbf{x}^*))$;
    if $\rho\left(\hat{\Phi}^*, \hat{\Phi}^{obs}\right) < \epsilon$ **then**
        retain $\hat{\Phi}^*$;
        set $i = i + 1$;
    end
end

population mutation rate of 1×10^{-6} mutations per site per individual per generation and a total effective breeding population size of 50,000 individuals at time 0. Samples were taken as follows; 75 modern sequences, 50 sequences from 1000 generations in the past and 50 sequences from 2000 generations in the past. For increased computational efficiency, our algorithm differs from the Algorithm 4 in that we do not produce simulations one at a time, and stop when R simulation have been accepted.

Instead, we use BayeSSC to produce $\Gamma^* >> \Gamma$ simulations, and retain the 'closest' $\alpha \times 100\%$ of these simulations, as defined by our distance metric $\rho(\cdot, \cdot)$. This avoids the need to repetitively request BayeSSC produce simulations one at a time.

The parameters and prior distributions for the population models used for our ObsDat, TrainDat and ABCDat data sets are summarised visually in Table 4.2, and are plotted in Figure 4.3.

The three models of population dynamics are;

Population Model	ObsDat	TrainDat	ABCDat
Constant	$N_e(0) = 5 \times 10^4$	$\Gamma^* = 5 \times 10^4$ $N_e(0) \sim U(5 \times 10^3, 1.5 \times 10^5)$	$\Gamma^* = 5 \times 10^4$ $N_e(0) \sim U(10^4, 10^5)$
Exponential	$N_e(0) = 5 \times 10^4$ $K = \frac{\ln(0.25)}{2000}$	$\Gamma^* = 1.5 \times 10^5$ $N_e(0) \sim U(5 \times 10^3, 1.5 \times 10^5)$ $K \sim U\left(\frac{\ln(0.1)}{2000}, \frac{\ln(0.9}{2000}\right)$	$\Gamma^* = 5 \times 10^4$ $N_e(0) \sim U(10^4, 10^5)$ $K \sim$ s.t. proportional decrease is $U(0.15, 0.75)$
Migration (Four constant and equal sized sub-populations)	For each sub-population: $N_e(0) = 1.25 \times 10^3$ Migration rate $= 0$	$\Gamma^* = 5 \times 10^4$ For each sub-population: $N_e(0) \sim U(12.5 \times 10^2, 3.75 \times 10^4)$ Migration rate $= 0$	$\Gamma^* = 5 \times 10^4$ For each sub-population: $N_e(0) \sim U(2.5 \times 10^3, 2.5 \times 10^4)$ Migration rate $= 0$

Table 4.2: Parameter values and prior distributions for all simulated population models.Common to all simulations: Sequence length $\ell = 1000$ bp, Mutation Model: Jukes-Cantor and a mutation rate $\mu = 10^{-6}$ per site per individual per generation.

1. A constant population model with an initial population size of 50,000 (the red dashed line).

2. An exponentially decreasing population (going backwards in time) with a decay rate in population size of $k = \frac{\ln(0.25)}{2000}$ and initial population size of $50,000$ (the green dashed line).

3. A 'migration model'. Here, four equal and constant sized sub-populations split from a common ancestral population of size $50,000$ at $150,000$ generations in the past, and have been unable to migrate (interbreed) since (the blue dashed line). We sample 13, 12, 12 and 13 sequences from sub-populations 1, 2, 3 and 4 respectively at 1000 generations in the past, 12, 13, 13 and 12 sequences from sub-populations 1, 2, 3 and 4 respectively at 2000 generations in the past, and 18, 19, 19 and 19 sequences from sub-populations 1, 2, 3 and 4 respectively at 0 generations in the past. We sample from the sub-populations in this way so that each sub-population is most evenly represented in the final temporal samples.

Initially we focus on the issue of parameter estimation, and we compare our estimates with those obtained from BEAST. For each ObsDat we performed a Bayesian Skyline Plot analysis in BEAST with the following settings.

1. Known tip dates for ancient sequences as per the sampling times.

2. A Jukes-Cantor mutation model (a GTR model with base frequencies 'all equal' and equal mutation rates).

3. A 'strict clock' with known mutation rate 1×10^{-6}.

4. 10 groups for generalised intervals under a piecewise-constant Skyline Model.

5. A chain length of 10,000,000.

All other prior distributions and operators are at default settings.

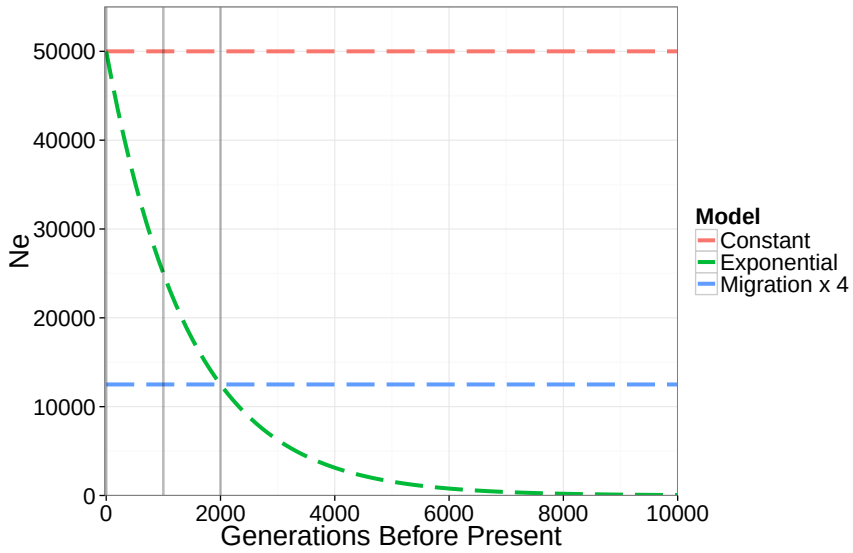


Figure 4.3: Simulated Model Dynamics with sampling times indicated by vertical grey lines. Note that the Migration Model population size is *per sub-population*.

4.3.1 Summary Statistics Used

To reproduce the output given by BayeSSC, we produce each of the single sample summary statistics on every separate group (the modern and both ancient samples), and then on a pooled sample, giving four sets of single sample summaries. One for the modern samples (Group 0), one for the ancient samples at 1000 generations before present (BP) (Group 1), one for the ancient samples at 2000 generations BP (Group 2), and one for the pooled sample of all sequences (Group 3). Three sets of multiple sample summary statistics are then made for Groups 0 and 1, Groups 0 and 2 and Groups 1 and 2. We refer to these seven summary statistic groupings as 'stat groups', but will refer to the three groups defined by the sampling times as 'temporal groups'.

In total, thirty-five candidate summary statistics are produced for each simulation.

4.3.2 Constant Model Analysis

Step 0: Obtaining ObsDat.

ObsDat for the Constant Model with an effective population size of 50,000 breeding individuals was randomly generated and treated as the known, real data.

Step 1: Producing TrainDat and defining the Linear Model

For the TrainDat for the Constant Model we simulated $50,000$ realisations of the Constant Model with a prior distribution of $N_e(0) \sim U(5000, 150000)$ for our training set. Using TrainDat we fitted a linear regression model to the training set on the transformed initial effective population size. Note that the initial effective population size $N_e(0)$ is analogous with the effective population size N_e in the constant population model.

The fitted linear model was of the form

$$\hat{N_e(0)}^{\lambda_{M_1}} = \beta_0 + \sum_{i=1}^{24} \beta_i s_i$$

where the observed Box-Cox transformation parameter (see Figure 4.4) for the Constant Model was $\lambda_{M_1} = 8.352 \times 10^{-2}$ (see Section A.3 for the list of the 24 retained summary statistics and associated coefficients), after backwards selection.

Summary statistics from each 'stat group' (the sets of samples which we pool together when calculating summary statistics) were retained as significant, and there appears to be no significant pattern to the retained summary statistics. Retained for each stat group were private alleles ($p_{v,w}$), Tajima's D-Statistic d_v (a measure of neutrality), nucleotide diversity π_v and the average gene flow ($F_{ST}^{(v,w)}$). However, at least one of each of the categories of summary statistics is retained as significant (see Figure 4.21).

Step 2: ABC comparing Obsdat to ABCDat

For the ABCDat we simulated $50,000$ realisations of the Constant Model with a prior distribution for $N_e \sim U(10^4, 10^5)$ and retained the 500 'closest' simulations

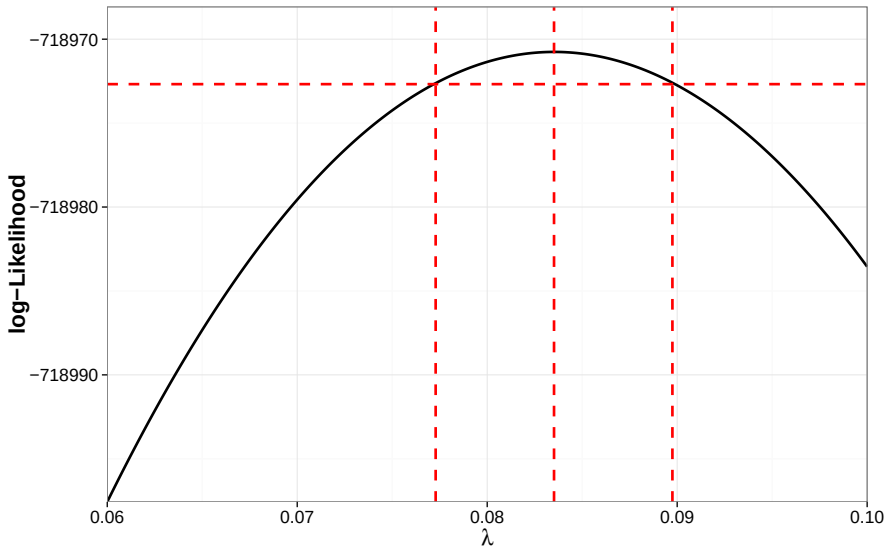


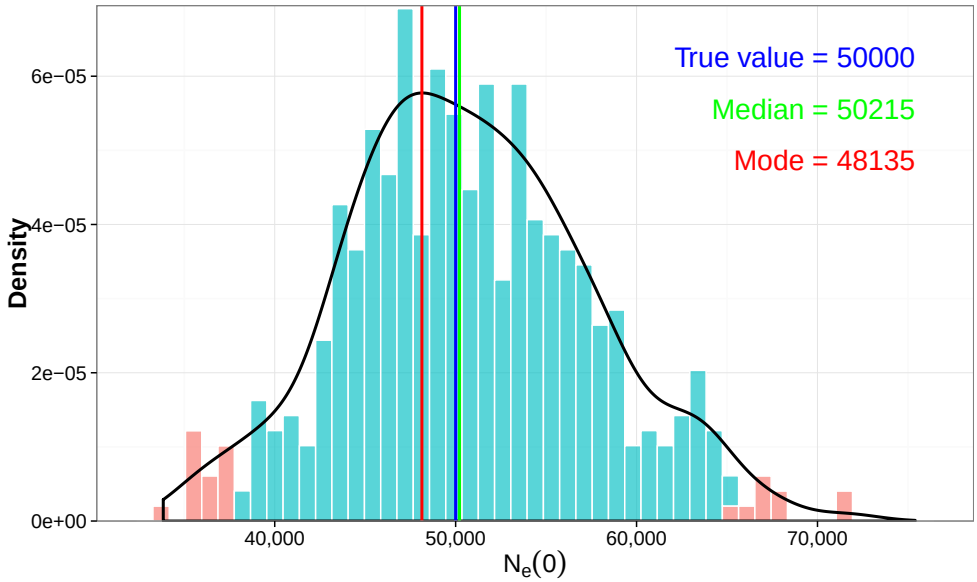
Figure 4.4: Box-Cox transformation profile likelihood with 95% confidence interval for λ_{M_1}.

for the untransformed posterior estimates.

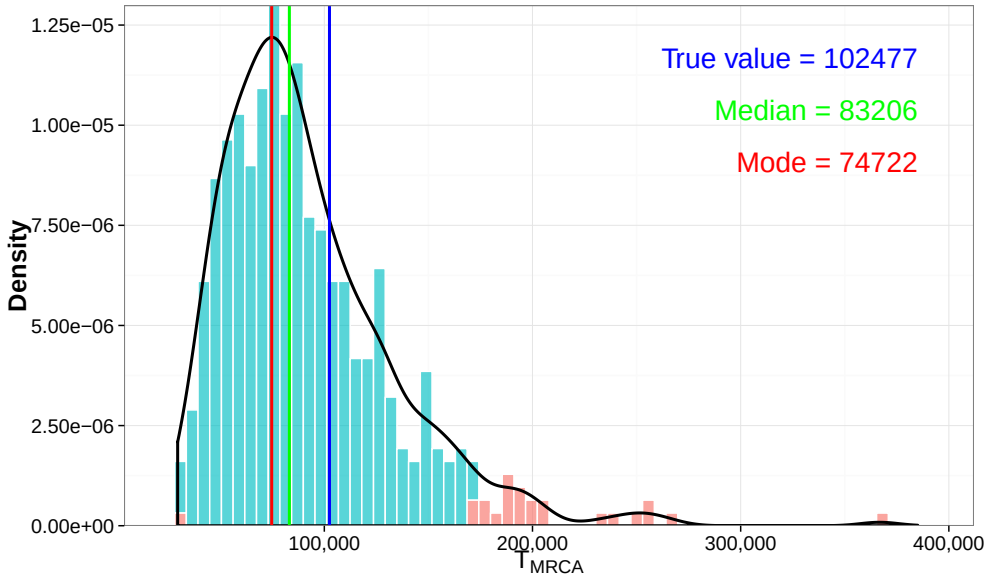
This led to an estimate of the median effective population size of $N_e = 50215$, with a 95% probability interval $(38082, 64800)$ (see Figure 4.5). The width of the 95% probability interval for the estimated effective population size is then 26718 breeding individuals. This compares favourably with the BEAST analysis performed on ObsDat. The BEAST analysis returned a 95% probability interval width for the estimated effective population size of at least 42799.37, and at most 129656.2 breeding individuals (see Figure 4.7).

Similarly, we estimate a T_{MRCA} of 83206 generations, with a 95% probability interval $(31030, 171115)$ (see Figure 4.5). Note that the BEAST analysis reports a 95% probability interval width for the T_{MRCA} of $(110260, 141570)$, which is narrower than our reported 95% probability interval. However, the BEAST probability interval does not contain the true T_{MRCA} for our data of 102477 generations (see Figure 4.6).

An independent sample of 10000 simulations of the Constant Model with $N_e(0) =$



(a)



(b)

Figure 4.5: Posterior samples for (a) the initial population size and (b) the T_{MRCA} for the Constant Population Model with the true value, the posterior mode of the kernel density estimate and the posterior median indicated. The 95% probability interval is highlighted light blue. The kernel density estimate is given in black.

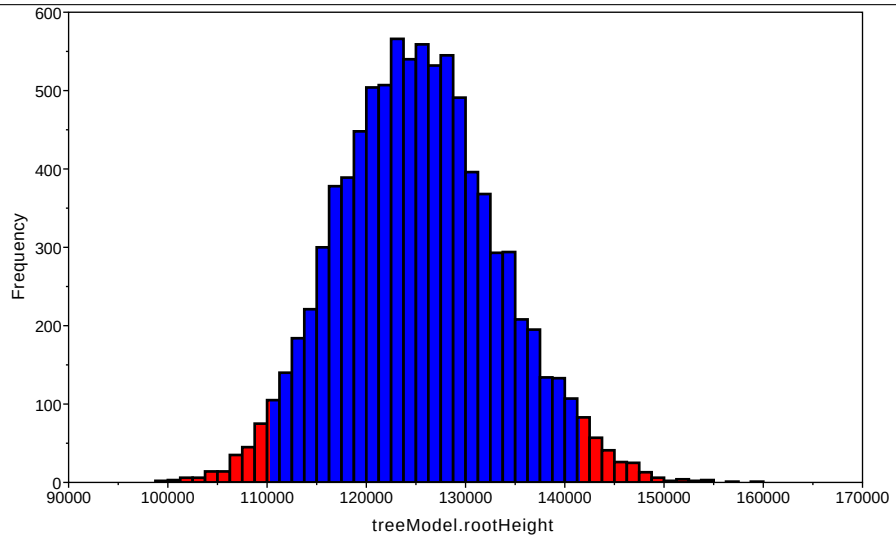


Figure 4.6: BEAST posterior sample for the $T_{\mathbf{MRCA}}$ from Tracer 1.5.

$50,000$ was produced. The values for the T_{MRCA} with parameters identical to Obsdat were recorded. This sample had a range of between 18188 and 726740 generations, and a 95% probability interval of $(26491, 206206)$. Clearly the T_{MRCA} has an extremely large variance, and hence our probability interval width is relatively small.

Finally a comparison of our findings and those produced via the BEAST analysis is shown in Figure 4.7. The red and blue solid lines are the posterior sample median estimates obtained via our method and BEAST respectively. Similarly, the red and blue shaded areas 95% probability regions obtained via our method and via BEAST respectively. The horizontal black dashed line is the true effective population size and the vertical dotted lines are the sampling times.

It can be seen that the population size dynamics, as reported by BEAST, below a few thousand generations before present are heavily driven by sampling events. We know that this was a constant sized population, yet we could easily infer a population crash at less then 2000 generation before present, followed by a population recovery around 1000 generations before present. These perceived dynamics from BEAST are probably driven by two things.

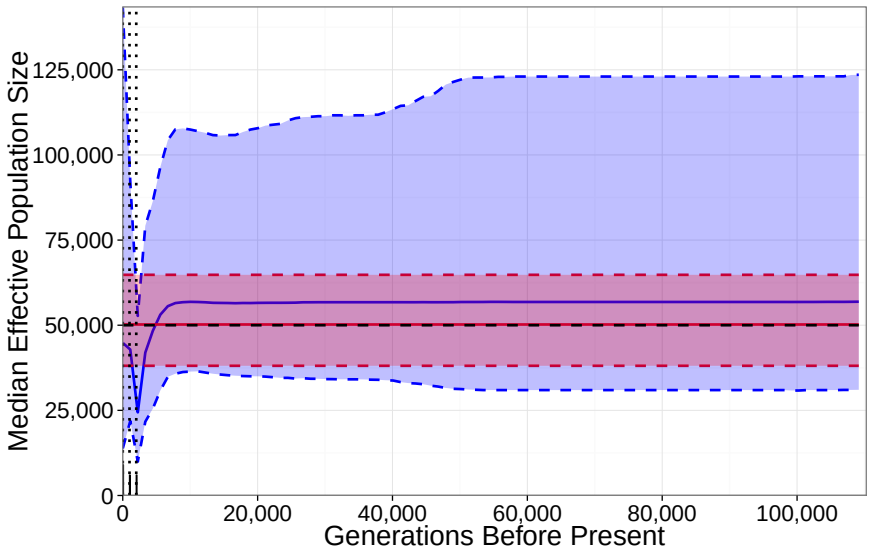


Figure 4.7: A comparison of our median effective population size estimates (in red) with those obtained from BEAST (in blue). The black dashed line is the true effective population size. The red and blue shaded areas are the 95% probability regions.

First, there is a prior belief that each harmonic mean population size over each successive interval is exponentially distributed with a mean of the previous interval. That is, each interval (from left to right in the BSP) obtains information from the previous intervals. Hence the intervals closest to zero generations in the past have less information with which to work, and thus have a greater variance.

Second, by ensuring that we had ten intervals into which we group our $(n+s-2 = 151)$ initial intervals, we force most of these intervals to occur at varying locations within $(0, 5000)$ generations before present (see Figure 4.8). This is due to the large intervals in the 'tail' of the BSP forcing the remaining grouped intervals to be very small. Because BEAST averages the harmonic mean effective population sizes over time for all MCMC iterations over the chain (and the Bayesian prior has no 'time awareness') these population size estimates vary most where coalescence is occurring rapidly, since N_e estimates are extremely sensitive to interval width (for small intervals).

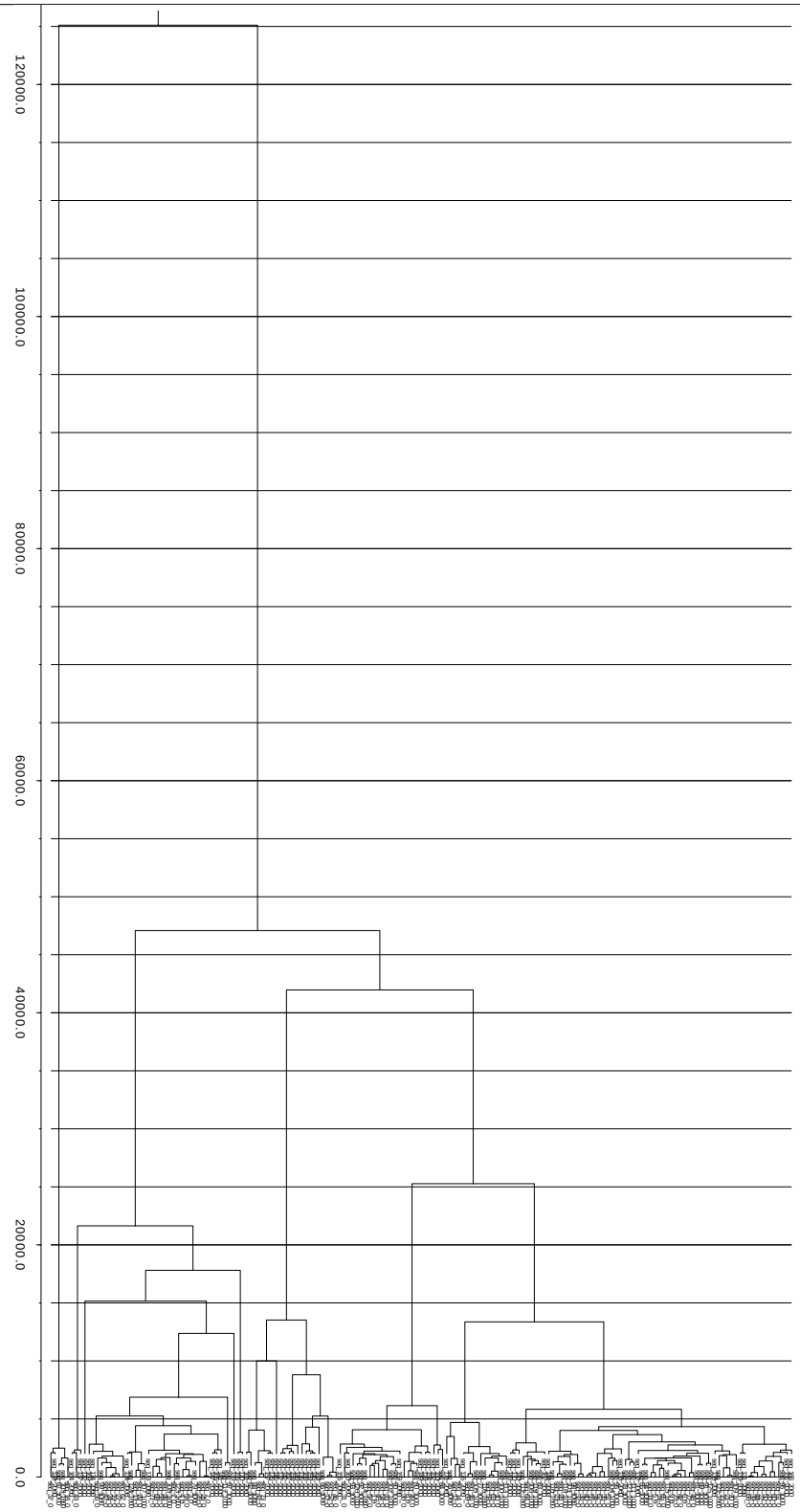


Figure 4.8: The annotated tree produced by the BEAST analysis of our Constant Model data.

4.3.3 Exponential Model Analysis

We begin by attempting to estimate parameters for the population size model

$$N(t) = N_0 e^{Kt}$$

where N_0 is the initial population size, and K is some decay rate for the effective population as we go backwards in time.

Step 0: Obtaining ObsDat.

ObsDat for the Exponential Model, with $N_e(0) = 50,000$ and $K = \frac{\ln(0.25)}{2000}$, was randomly generated and treated as the known, real data.

Step 1: Producing TrainDat and defining the Linear Model

For the TrainDat we simulated $150,000$ realisations of the Exponential Model with a prior distribution for $N_e(0) \sim U(5000, 150000)$, and $K \sim U(\ln(0.1)/2000, \ln(0.9)/2000)$. The prior distribution for K corresponds to a 10% to 90% decrease in population size at 2000 generations in the past, but the proportional decrease is not uniformly distributed under this interpretation.

Using the TrainDat we fitted the linear regression models to the transformed effective population size.

The fitted linear model for the initial population size $N_e(0)$ was of the form

$$\hat{N_e(0)}^{\lambda^1_{M_2}} = \beta_0 + \sum_{i=1}^{27} \beta_i s_i$$

where the observed Box-Cox transformation parameter (see Figure 4.9) for the Exponential Model was $\lambda^1_{M_2} = 6.8392 \times 10^{-2}$ (see Section A.4 for the list of the 27 retained summary statistics and associated coefficients), after backwards selection.

Summary statistics from each 'stat group' were retained as significant, and like the Constant Model analysis there appears to be no significant pattern to the retained summary statistics. Retained for each stat group were Tajima's D-Statistic d_v (a measure of neutrality), nucleotide diversity π_v, pooled haplotype diversity H_T,

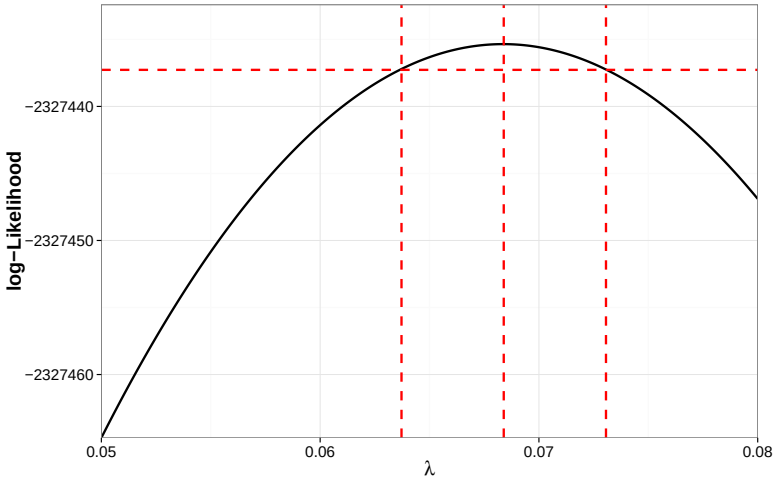


Figure 4.9: Box-Cox transformation profile likelihood with 95% confidence interval for $\lambda^1_{M_2}$.

segregating sites S_v and the average gene flow ($F_{ST}^{(v,w)}$). Again, at least one of each category of summary statistics was retained as significant.

The fitted linear model for the population size decay rate K was of the form

$$\hat{K}^{\lambda^2_{M_2}} = \beta_0 + \sum_{i=1}^{27} \beta_i s_i$$

where the observed Box-Cox transformation parameter (see Figure 4.10) for the Exponential Model was $\lambda^2_{M_2} = 1.7257$ (see Section A.5 for the list of the 27 retained summary statistics and associated coefficients), after backwards selection.

The retained summary statistics are largely the same as for the linear model for estimating the initial population size for the Exponential model, with a few exceptions. The linear model for the rate K makes use of the number of alleles private to group 1 vs group 2, $p_{(1,2)}$, and the mean haplotype diversity for groups 1 and 2, $\bar{H}_s^{(1,2)}$, where the linear model for the initial effective population size did not. However, the linear model for the rate K does not retain the pooled haplotype diversity in group 0, $H_t^{(0,1)}$, and the nucleotide diversity in group 3, $\pi_{\{0,1,2\}}$, as significant.

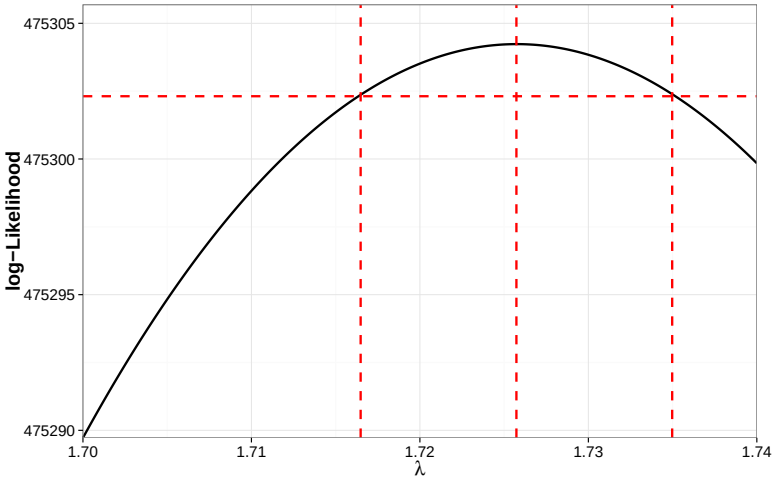


Figure 4.10: Box-Cox transformation profile likelihood with 95% confidence interval for $\lambda^2_{M_2}$.

Step 2: ABC comparing Obsdat to ABCDat

For the ABCDat we simulated $150,000$ realisations of the model with a prior distribution for $N_e(0) \sim U(10^3, 10^4)$, and K is distributed such that the proportional decay in population size 2000 generations in the past has distribution $U(0.15, 0.75)$ and retained the 500 'closest' simulations (as defined by the Euclidean Distance in Section 4.2.3) for the untransformed posterior estimates.

We estimate a median initial effective population size of $N_e(0) = 49872$, with a 95% probability interval $(28251, 79463)$ (see Figure 4.11), and a median population decay rate of $K = -6.7474 \times 10^{-4}$, with a 95% probability interval of $(-9.2587 \times 10^{-4}, -4.1082 \times 10^{-4})$ (see Figure 4.11). The posterior estimates for the decay rate K are associated with an estimated median reduction in population size to 25.94% of the initial population size, with a 95% probability interval of $(15.70, 43.97)$ % for the reduction.

By fitting an Exponential Model of the form $N_e(t) = Ae^{Kt}$ to the BEAST analysis such that the estimated N_e line was not outside of the 95% probability limits, we were able to find upper and lower bounds on an estimate for K (see Figure

4.12). It is important to note that it is not the case that all paths drawn inside the 95% probability intervals presented are equally credible. Recall that when BEAST traverses tree space, and returns population size estimates based on the accepted genealogies, the variation associated with the coalescence events is not considered (see Section 3.1.3). Hence, when BEAST returns a BSP, we show only variation in $N_e(t)$ (the y-direction), and not in time (the x-direction). However, we fit Exponential Model curves to the BEAST data for a basis for comparison as we can not retrieve any sensible estimates of K from BEAST directly (for a full discussion on this topic, see Section 6.3).

The minimum and maximum values for K from our BEAST analysis are -12.92×10^{-4} -3.14×10^{-4} respectively. These correspond to proportional decreases in effective population size of 7.54% and 53.41% respectively. The initial effective population size associated with these lower and upper estimates were 24839.94 and 68132.71 respectively.

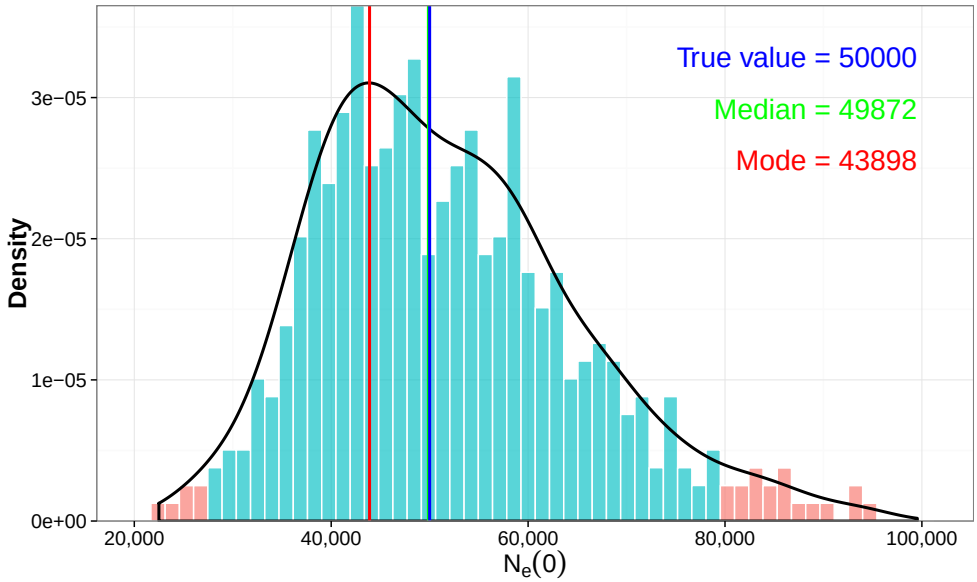
We estimate a T_{MRCA} of 6074 generations, with a 95% probability interval $(4543, 8069)$ (see Figure 4.13). The BEAST analysis reports a very similar median T_{MRCA} of 5758 generations and 95% probability interval width for the T_{MRCA} of $(4007, 8016)$. Both intervals contain the ObsDat T_{MRCA} for our data of 5483 generations.

Finally a comparison of our findings and those produced via the BEAST analysis is shown in Figure 4.14. Recall from Section 3.1.3 that BEAST produces a BSP for each posterior genealogy sampled. BEAST then takes a sample of estimates of $N_e(t)$ for each

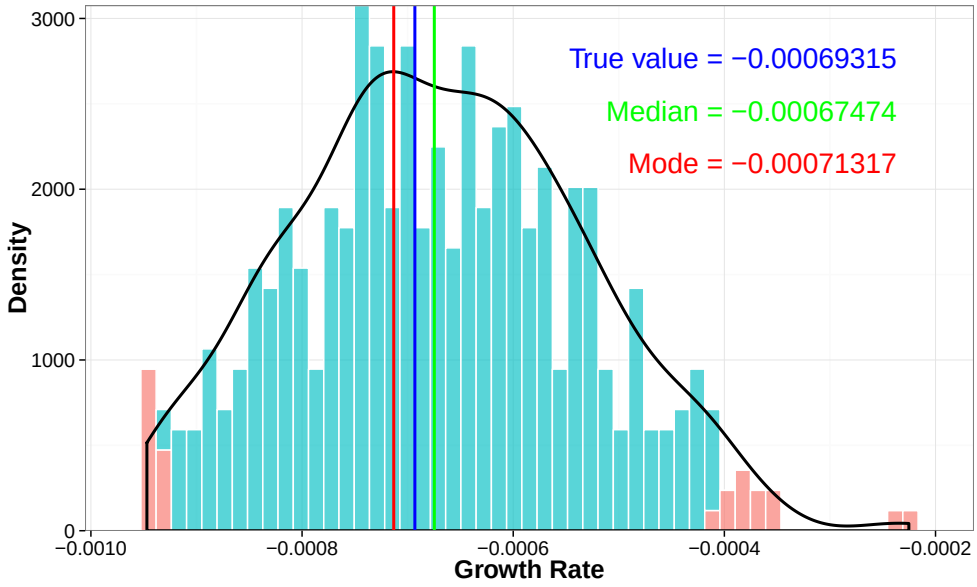
$$t \in \{0, \frac{T_{lower}}{99}, 2 \times \frac{T_{lower}}{99}, 3 \times \frac{T_{lower}}{99}, \cdots, 98 \times \frac{T_{lower}}{99}, T_{lower}\}$$

where T_{lower} is the the lower limit for the 95% probability interval for T_{MRCA}. BEAST plots the median and the lower and upper limits of the 95% probability interval of $N_e(t)$ at each t.

As a fair comparison, we do a similar thing when we plot our results with the



(a)



(b)

Figure 4.11: Posterior samples for (a) the initial population size and (b) the population decay rate for the Exponential Population Model with the true value, the posterior mode of the kernel density estimate and the posterior median indicated. The 95% probability interval is highlighted light blue. The kernel density estimate is given in black.

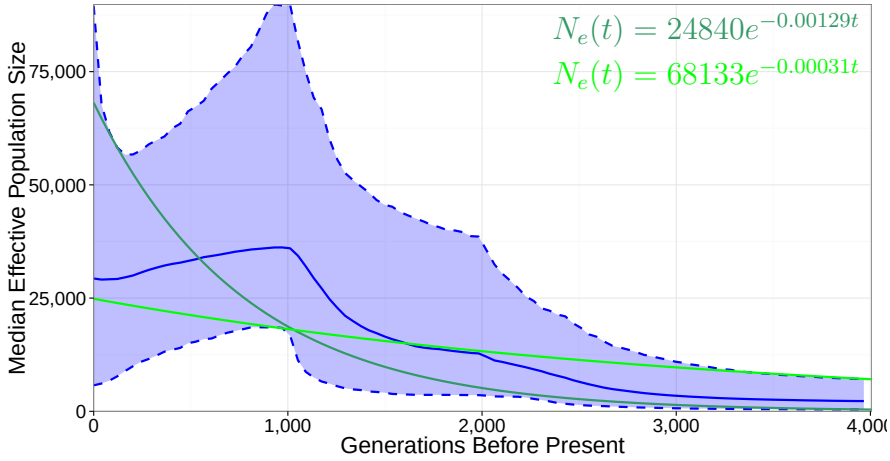


Figure 4.12: Estimates of the largest and smallest population decay rates (the green lines) for a fitted Exponential Model from the BEAST analysis. The blue solid line and shaded area are the posterior sample mean and the corresponding 95% probability interval for $N_e(t)$.

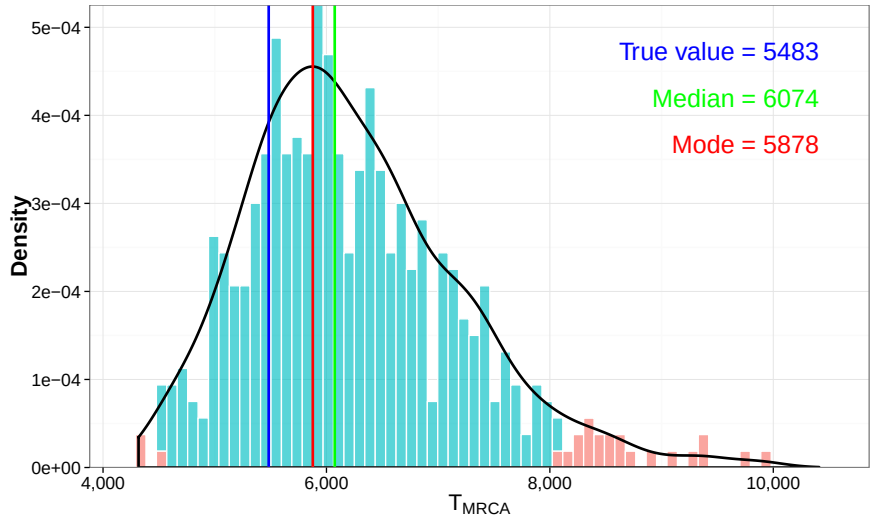


Figure 4.13: Posterior samples for T_{MRCA} for the Constant Population Model with the true value, the posterior mode of the kernel density estimate and the posterior median indicated. The 95% probability interval is highlighted light blue. The kernel density estimate is given in black.

results from BEAST. For each of the retained simulations in our posterior sample, we reconstruct the population size dynamics, and then record the population size estimates at the same times as BEAST. We then plot the median and the lower and upper limits of the 95% probability interval of $N_e(t)$ at each of these times.

For this reason, when we compare our analysis to the corresponding BEAST analysis visually, the recovered population dynamics for our analyses may not strictly represent the individual parameter estimates. In the case of the single parameter, constant sized population models (the Constant Model and the Migration Model), if we use the plotting scheme BEAST uses, or if we just plot the median and the lower and upper limits of the 95% probability interval of $N_e(t)$ as solid and dashed lines respectively, the plots would be identical.

However, note that Figure 4.14 is produced in the same way that BEAST produces its BSPs. Hence, the figure does not show the most extreme values of the 95% probability intervals for $N_e(0)$ and K together. That is, we do not include a population dynamic where $N_e(0) = 28251$ *and* $K = -4.1082 \times 10^{-4}$, even though these values are in the 95% probability intervals for the parameters respectively.

Both our analysis and the BEAST analysis perform comparably when we consider only the parameter values, and importantly, both probability intervals contain the true effective population size $N_e(t)$ for all t. However, when we compare the BEAST BSP and our results produced in the same way, we immediately observe that our analysis has produced estimates with noticeably smaller interval widths. It is important to keep in mind that our analysis assumes an Exponential Model, and hence the shape of our estimates will inherently follow an exponential curve.

We again make note of the effect of sampling times on both the median effective populations size estimates, and the 95% probability interval bounds, for the BEAST analysis. On viewing the BEAST analysis, we might be tempted to infer a steadily increasing effective population size since 4000 generations before present, until roughly 1000 generations before present, followed by a steady decline in ef-

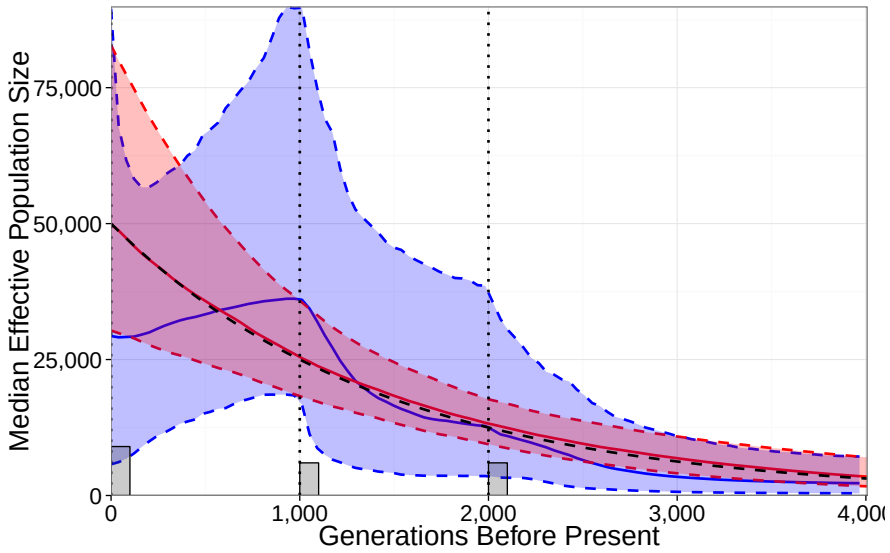


Figure 4.14: A comparison of our median effective population size estimates (in red) with those obtained from BEAST (in blue). The black dashed line is the true effective population size.

fective population size. This decline in effective population size reduces the lower bound on the effective population size to be as small as 5737 breeding individuals at 'present', a time at which we know the true population size to be 50,000 breeding individuals.

We include the annotated tree produced from the BEAST analysis for comparison (see Figure 4.15). The relatively wide spread of coalescent events leads to a reasonably even interval width for the 95% probability interval for the effective population size for the corresponding proportion of the BSP. However, the sampling events still have a strong effect on the recovered BSP.

4.3.4 Migration Model Analysis

The Migration Model data is the only one of the models that violates the assumption of panmixia. In the Migration Model, we allow four separate sub-populations to evolve independently of one another for $150,000$ generations, with no migration

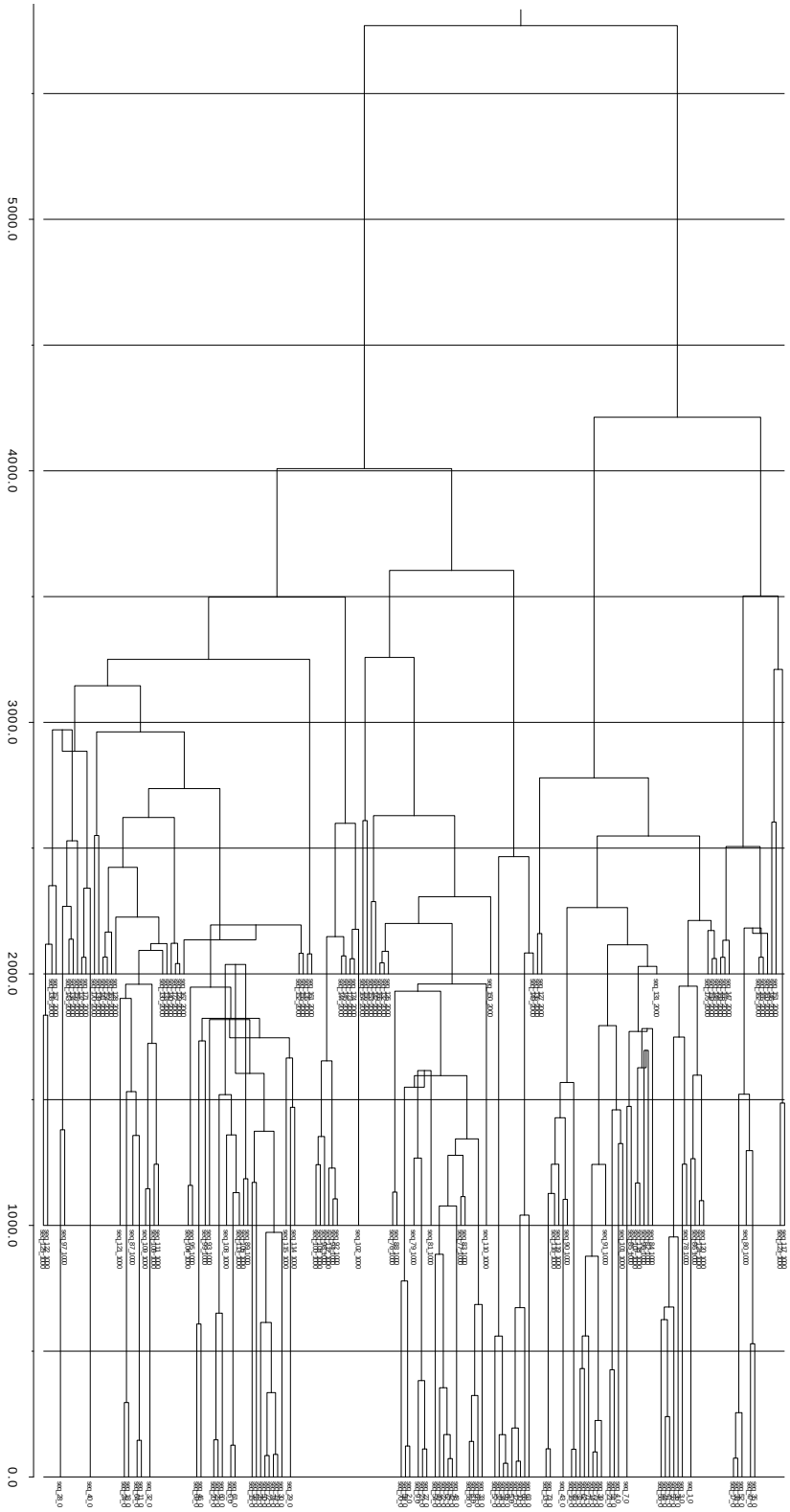


Figure 4.15: The annotated tree produced by the BEAST analysis for our Exponential Model data.

between the sub-populations. We call this the 'migration model', although we set our rate of migration to zero. By having isolated sub-populations, we tacitly allow more than the usual number of alleles to become fixed (high in proportion) leading to an inflated estimate of genetic diversity.

For the analysis of the data we estimate only one initial effective population size, and make the four sub-populations of equal size. We do this to avoid having to estimate four parameters for the purpose of model comparison in the next chapter. The estimated *total* effective population size estimates are then the posterior estimates multiplied by four.

Step 0: Obtaining ObsDat.

ObsDat for the Migration Model, with four sub-populations of constant size 12,500 breeding individuals (incapable of migration), was randomly generated and treated as the known, real data.

Step 1: Producing TrainDat and defining the Linear Model

For the TrainDat we simulated $50,000$ realisations of the model with a prior distribution for $N_e(0) \sim U(1250, 37500)$. Using the TrainDat set we fitted a linear regression model to the training set on the transformed effective population size.

The fitted linear model was of the form

$$\hat{N_e(0)}^{\lambda_{M_3}} = \beta_0 + \sum_{i=1}^{19} \beta_i s_i$$

where the observed Box-Cox transformation parameter (see Figure 4.16) for the constant model was $\lambda_{M_1} = 1.4904 \times 10^{-1}$ (see Section A.3 for the list of the 19 retained summary statistics and associated coefficients), after backwards selection.

This time, the retained summary statistics seemed to display some pattern. Tajima's D-statistic and the number of segregating sites were retained for every stat group. The nucleotide diversity was retained for nearly every stat group (only π_1 was not retained), and then only the number of modern haplotypes (h_0), and the mean haplotype diversity between groups 0 and 1 ($\bar{H}_s^{(0,1)}$) were retained. Haplotype

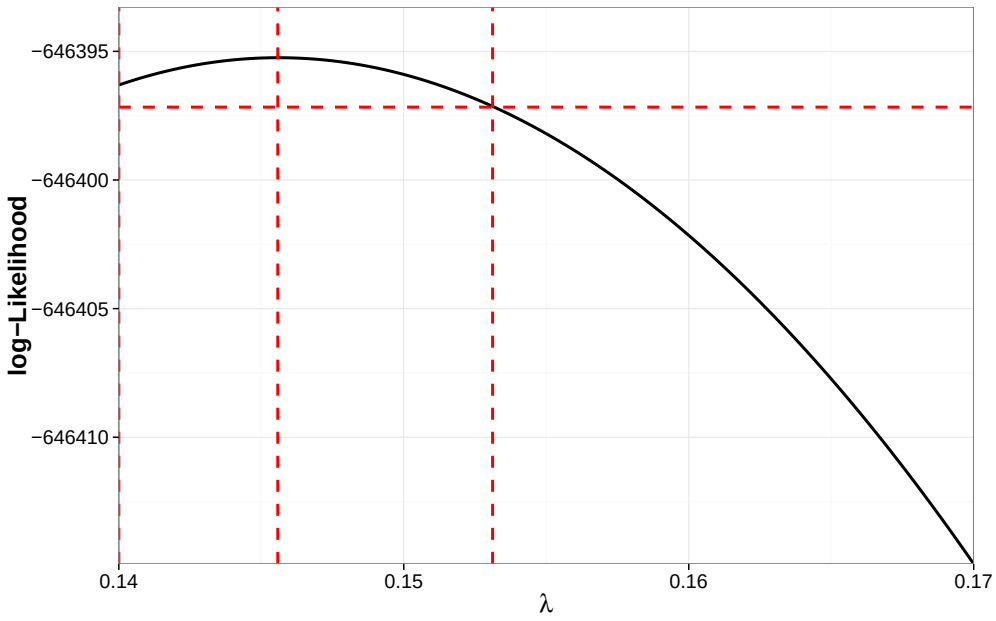


Figure 4.16: Box-Cox transformation profile likelihood with 95% confidence interval for λ_{M_3}.

diversity and gene flow were never retained as significant.

Step 2: ABC comparing Obsdat to ABCDat

For the ABCDat we simulated $50,000$ realisations of the model with a prior distribution for $N_e(0) \sim U(2500, 25000)$ and retained the 500 'closest' simulations for the untransformed posterior estimates.

We estimate a median effective population size (for each sub-population) of $N_e(0) = 13373$, with a 95% probability interval $(10200, 16697)$ (see Figure 4.17). This corresponds to a total population size estimate of 53492 breeding individuals, and a 95% probability interval of $(40376, 66788)$ total breeding intervals.

From the BEAST analysis (see Figure 4.18) we see two things. First, the estimated median effective population size is 105586.78 breeding intervals from 26514 to 128382 generations in the past. The 95% probability interval during this time period does not contain the true effective population size. This inflated effective population size estimate is a result of the violation of panmixia inferring a much

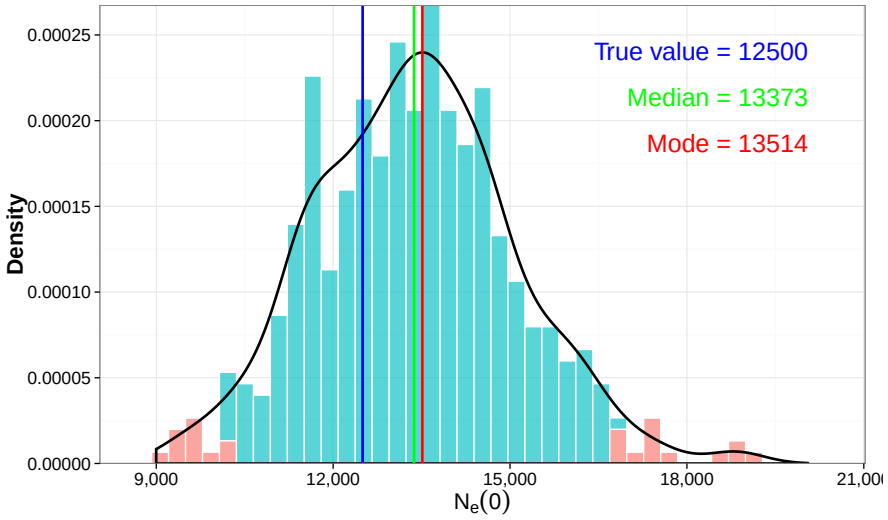


Figure 4.17: Posterior samples for the effective population size N_e for the Migration Population Model with the true value, the posterior mode of the kernel density estimate and the posterior median indicated. 95% probability interval is highlighted light blue. The kernel density estimate is given in black.

higher genetic diversity than is actually present.

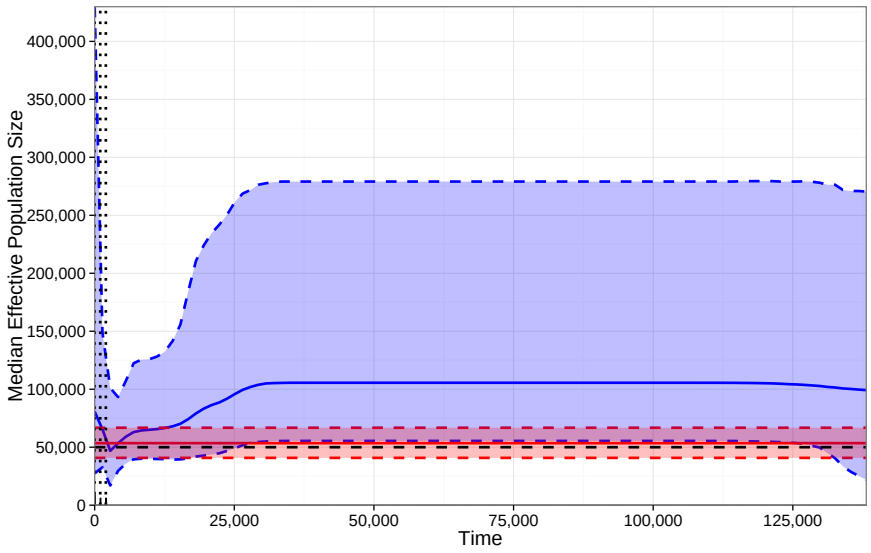


Figure 4.18: A comparison of our median effective population size estimates (in red) with those obtained from BEAST (in blue). The black dashed line is the true effective population size.

Second, if we look at the reconstructed genealogical tree from BEAST (see Figure 4.19), we see that almost all coalescence has occurred at 25000 generations in the past, and that the final four sub-populations do not coalesce until approximately $125,000$ generations in the past. This lack of coalescence events leads to a lack of information around that time, and hence the interval width of 223692.7 is relatively large.

Again we notice effective population size dynamics being driven by sampling times, and hence we see spurious inferences around these times. The population size decline between $25,000$ and 2000 generations in the past, followed by a sharp population size increase until generation zero is a result of coalescent event activity. Recall that the true population size is a constant $50,000$ breeding individuals.

We estimate a T_{MRCA} of 166091, with a 95% probability interval (150445,194996) (see Figure 4.20). The BEAST analysis reports a 95% probability interval of

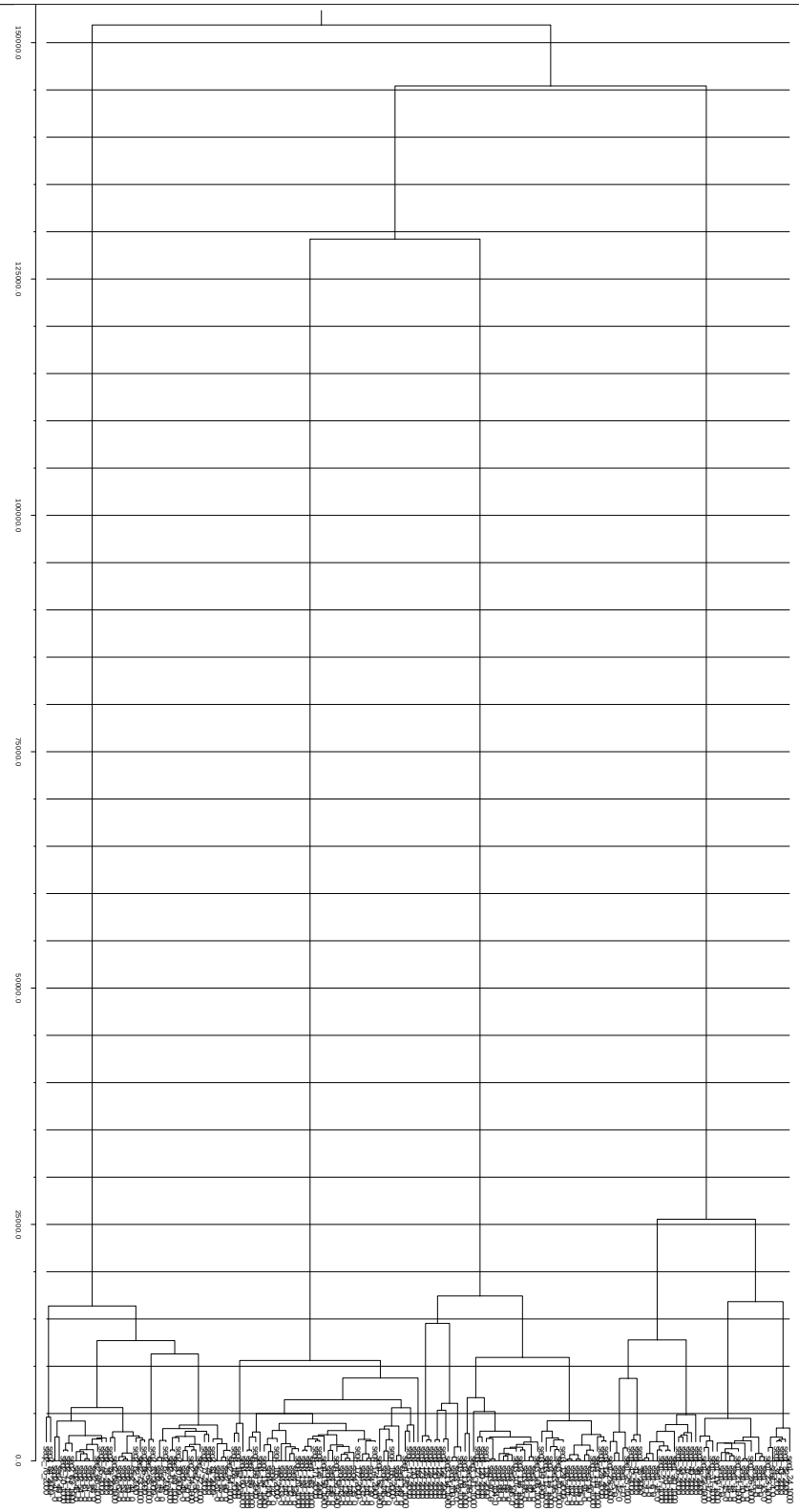


Figure 4.19: The annotated tree produced by the BEAST analysis of our Migration Model data.

(138150,165620) for the T_{MRCA}, with posterior median 151870 generations.

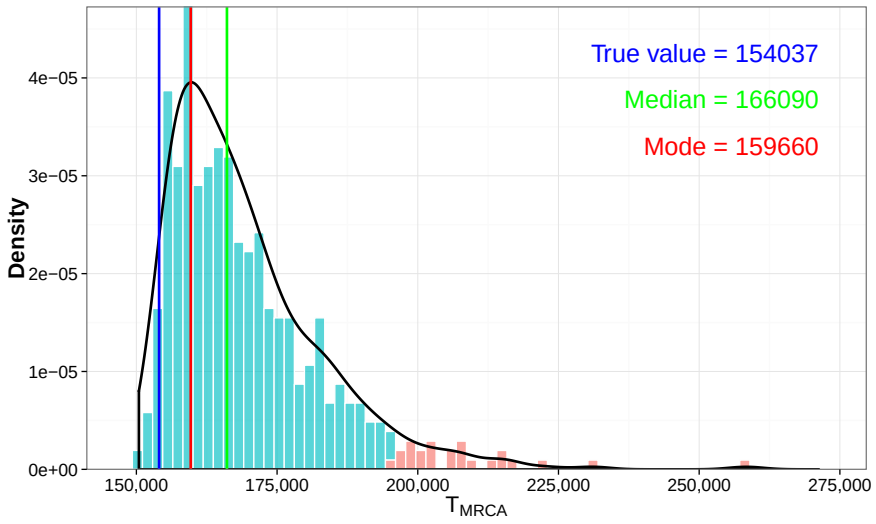


Figure 4.20: Posterior samples for T_{MRCA} for the Migration Population Model with the true value, the posterior mode of the kernel density estimate and the posterior median indicated. The 95% probability interval is highlighted light blue. The kernel density estimate is given in black.

Due to the nature of how we specify the simulation model in BayeSSC, we must choose a time when a single population splits into four sub-populations. In specifying this 'split time', we impose a $T_{\text{MRCA}} > 150000$. BEAST estimates this quite accurately, and there is no reason that estimation of this parameter can not be added to an ABC scheme via our method.

4.3.5 Parameter Estimation Comparisons

Each of the parameter estimation analyses managed to construct probability intervals containing the true initial effective population size. Specifically, we managed to construct probability intervals of lesser width than those of the corresponding BEAST analyses, that contained the known effective population size at all times. In the case of the Exponential Model we also managed to estimate the popula-

tion growth rate with a closer upper and lower estimates than the corresponding BEAST analysis might infer. It is important to note that we compare the results given by Beast and our method for only three analyses.

However, we concede that in all of the above analyses, we specify a model of population dynamics, and then simulate under this model, with a number of known parameters. We then specify sensible prior distributions for the parameters of interest, ($N_e(0)$ and K). This was never more apparent than when we simulated the four sub-populations of *equal size* and *known split time* from the single population 150,000 generations in the past. On the other hand, BEAST has a practically uninformative prior ($U(0, 10^{100})$) for the effective population size, and in the case of the Exponential Model, had no concept of K built into the analysis.

The BEAST analysis has no 'known model' luxury, and must attempt to reconstruct the population dynamics using only the assumption of panmixia within a single breeding population. In real analyses these assumptions are rarely met and strong biases, as observed in the case of the Migration Model data, can result. This violation of panmixia introduced a bias that our parameter estimation method did not suffer from.

An attractive characteristic of a BEAST analysis is the ability to employ the analyses with very little prior information. However, in the presence of some prior knowledge about the population of interest, it seems wise to use this knowledge. Hence, if we can justify a candidate model of population dynamics (Constant, Exponential or Migration say), we can justify employing this population model in the ABC simulation design. To this end, in the next chapter we will explore methods of data driven model selection and comparison (albeit from a candidate set of population models).

We begin by discussing a classic method of model comparison called Bayes factors, and discuss common problems with any Bayesian methodology concerned with model selection or comparison. We then describe a fundamental problem

with these methods, even when ABC methodologies employ sufficient summary statistics.

We end by noting that our linear models for estimating posterior means share common summary statistics as significant predictors, and importantly, retain some summary statistics unique to certain models (see Figure 4.21). Hence, we suggest a method of data driven model classification that uses the information in simulated summary statistics via multinomial logistic regression.

Figure 4.21: Retained summary statistics for each transformed parameter linear model.

Chapter 5

Data Driven Model Selection

In Chapter 4 we described a method for parameter estimation using semi-automatic summary statistics and ABC. We showed that this method yielded probability intervals containing the correct parameter values, and that these intervals were of lesser width than those of the corresponding BEAST Skyline Plot analyses. To justify the use of this method for parameter estimation, we must justify the selection of the model from which we simulate our TrainDat and ABCDat data sets.

In this chapter we introduce Bayes Factors, a common method for post-hoc model comparison. We calculate Bayes Factors for each of the ObsDat (defined in Chapter 4), and highlight a key issue for any model comparison where the probability of a model, given an observed summary statistic, is estimated via posterior ratios of retained model observations.

We then introduce Multiple Logistic Regression, and describe how this can be used as a classification method for our ObsDat, via supervised learning. As a corresponding visualisation tool, we introduce Principal Component Analysis, and describe a possible method for identifying and supporting sensible classifications for a given ObsDat under a Multiple Logistic Regression methodology.

We apply these methods to the ObsDat we analysed in Chapter 4, and report our

results.

5.1 Common Problems and Risks for ABC Inference and Model Comparison

Bayesian frameworks lend themselves to intuitive and natural model selection methods [35]. While we employ a specific method of model comparison in the following section, called Bayes Factors, we first describe the common sources of error in ABC model comparisons.

1. Summary statistics are used to reduce the dimensionality of the data, and hence increase the acceptance rate of simulated observations. Low dimensional and sufficient summary statistics are uncommon in situations where ABC is used, and heuristic methods are employed to choose a 'best' set of summary statistics [8]. A poor choice of summary statistic can lead to inflated probability intervals. These inflated intervals are due to little information about the parameters of interest being available from the summary statistic when a comparison to the ObsDat is made. We aim to reduce the impact of this source of error using the algorithm for semi-automatic summary statistics, outlined in Section 4.

2. While not specifically a model selection issue, the choice for the tolerance parameter ϵ is not well-defined. Recall that ϵ is the preselected maximum distance our simulated data can be from our observed data before it is discarded. Clearly, our estimated posterior sample becomes an exact posterior sample when $\epsilon = 0$. However, the closer we set ϵ to zero, the more computationally intractable our simulation scheme becomes. Conversely, the larger ϵ, the larger the bias we introduce into our posterior sample [4]. Classically, sensitivity analyses are applied to the posterior distribution for varying val-

ues of ϵ [36]. Similarly, when comparing distances for models with parameter spaces of varying dimensions, the form of the distance metric might change. That is, a distance of ρ in a one-dimensional parameter space can be quite different from the same units of distance in a five-dimensional parameter space, say. Hence, the meaning of ϵ might be different for each model, and this may lead to spurious comparisons of 'closeness' for inter-model simulations. For our analyses we select ϵ such that we retain only the 500 'closest' simulations, and perform sensitivity analyses.

3. In only specifying a small number of models, we risk not properly exploring the hypothesis space (of all possible models) [36]. That is, in an effort to minimise computational cost, we must select from a small number of models *a priori*. Any analysis must rely on expert opinion and prior knowledge. Since we know the model under which our data was produced, we avoid this common pitfall.

4. The most obvious problem is that of the choice of prior distributions. Posterior distributions may be so sensitive to the choice of prior distributions that model selection is meaningless [44]. If one model has a better defined prior distribution for its parameters, then it should, more often, produce favourable results, and this will be represented in a favourable Bayes Factor (see Section 5.1.1) for the specific model [5]. Commonly, sensitivity analyses for the Bayes Factors for posterior distributions are employed.

5. Finally, the 'Curse of Dimensionality' refers to a need to increase the number of simulations as we increase the number of parameters we are estimating. It can be imagined that the more complex the system, the larger the unknown number of parameters, hence the more exploration of the space that is required. While large dimensional parameter spaces present issues for parameter inference, they also produce issues when comparing models of unequal parameter space dimension.

In our examples we present two one-dimensional parameter space models ($N_e(0)$ for the Constant and Migration Models), and a two-dimensional parameter space model ($N_e(0)$ and the growth rate K for the Exponential Model). For d simulations in the one-dimensional parameter space, we would like to explore the two-dimensional parameter space with d^2 simulations. This is beyond the scope of our project, so in an effort to avoid any issues when estimating parameters for the Exponential Model, we simulate three times as many simulated observations for both the training and ABC data sets.

5.1.1 Bayes Factors For Model Comparison

One of the most commonly used methods of model comparison is Bayes Factors [21]. Consider a candidate list of models $\mathcal{M} = \{M_1, \ldots, M_q\}$ with corresponding probabilities of selection under a sampling scheme, $R(M_i)$. The posterior probability of a model M_i, given some data $\mathbf{X}$ is then,

$$P\left(M_i \,|\mathbf{X}\right) = \frac{P\left(\mathbf{X} \,|M_i\right) R\left(M_i\right)}{P(\mathbf{X})}.$$

For each simulation, we begin by randomly selecting a model of population dynamics, simulate a realisation of that model, and continue the ABC scheme as before.

We aim to calculate the Bayes Factor for models i and j,

$$\begin{aligned} B_{i,j} &= \frac{P\left(\mathbf{X} \,|M_i\right)}{P\left(\mathbf{X} \,|M_j\right)}. \\ &= \frac{P\left(M_i \,|\mathbf{X}\right)/R\left(M_i\right)}{P\left(M_j \,|\mathbf{X}\right)/R\left(M_j\right)}. \end{aligned}$$

To find B_{ij}, we calculate the posterior ratio of the models i and j from our posterior

sample [42] as,

$$\frac{P(M_i|\mathbf{X})}{P(M_j|\mathbf{X})} = \frac{P(\mathbf{X}|M_i)\,R(M_i)}{P(\mathbf{X}|M_j)\,R(M_j)} = B_{i,j}\frac{R(M_i)}{R(M_j)}.$$

In practice, we calculate $B_{i,j}$ in the following way. We find the posterior ratio for models i and j, which is the number of simulations in the posterior sample that are from model i, divided by the number of simulations in the posterior sample that are from model j,

$$\frac{P(M_i|\mathbf{X})}{P(M_j|\mathbf{X})}.$$

We then multiply this by the inverse ratio of the probability of selecting model i, over the probability of selecting model j,

$$\frac{1/R(M_i)}{1/R(M_j)} = \frac{R(M_j)}{R(M_i)}.$$

5.2 Results for Bayes Factors

Recall the names given to our various data sets. ObsDat is the observed data (collection of sequences) we wish to analyse, TrainDat is the simulated data we use to train the linear models (the semi-automatic summary statistics) and ABCDat is the simulated data with which we compare ObsDat to obtain the posterior sample.

For each ObsDat (simulated under the Constant, Exponential and Migration Models), we perform the following steps;

1. Fit three transformed linear models for each of the three population dynamic models (the Constant, Exponential and Migration models), as in Section 4.2, using three sets of TrainDat.

2. Simulate three sets of ABCDat for each of the three population dynamics models (the Constant, Exponential and Migration models), and calculate distances between the *untransformed* posterior mean estimates of ObsDat and the three ABCDats.

3. Combine the three sets of distances between ObsDat and the three ABCDats.

4. Record the model under which the 'closest' 500 retained sampled parameters was simulated.

For each ObsDat we can claim which model we believe the data is generated under (if any model returns a significant Bayes Factor) as we know the true model is one of the three candidate models. Normally this is not the case, and one can only argue which model is most likely to have generated the data, out of the set of candidate models.

Throughout we will use the interpretation of the Bayes Factor as given by Kass and Raftery [18] (see Table 5.1).

$\mathbf{B_{i,j}}$	**Strength of Evidence for Model i over Model j**
1 to 3	Not worth mentioning
3 to 20	Positive
21 to 150	Strong
> 150	Very Strong

Table 5.1: The scale for interpretation of Bayes Factors as given by Kass and Raferty [18].

It is useful to keep in mind that $B_{i,j}$ is the strength of evidence in support of Model i, over Model j, given the data. Hence, if $B_{i,j} \to \infty$, then there is no evidence in

favour of Model j since

$$B_{j,i} = \frac{1}{B_{i,j}} = 0.$$

Similarly, if $B_{i,j} < 1$, and we wish to interpret this via Table 5.1, we simply consider $B_{j,i}$ (since $B_{j,i} > 1$).

5.2.1 Bayes Factors for the Constant Model Data

For the Constant Model ObsDat, we produce a table of Bayes Factors (see Table 5.2), and a histogram of the distances, ρ, for the 500 closest retained samples (see Figure 5.1).

Of the 500 closest simulated data sets, 240 were generated under the Constant Model, 0 were generated under the Exponential Model, and 260 were generated under the Migration Model.

We read Table 5.2 in the following way. Each cell is the strength of evidence for the ObsDat being the result of the model in the row, when compared to the model in the column. Take the cell in row 1 and column 3. This cell represents the strength of evidence for the data being a result of a Constant Model versus a Migration Model. Here we find that $B_{1,3} = 0.92$, hence we consider $B_{3,1}$, which is $1/B_{1,3} = 1.08$, and conclude there is no evidence worth mentioning in favour of the Migration Model over the Constant Model, and vice-versa.

It is important to consider that in practice, we do not know the true population model, and hence every cell of the table must be considered. Note that the j^{th} diagonal cell represents the strength of evidence for the j^{th} model over the j^{th} model, which will always be 1, and hence is never presented (a hyphen is given for ease of reading).

From Table 5.2 we would infer that the Constant Model and the Migration Model are equally likely models, but that the Exponential Model is infinitely less likely than either of these models, given our data. From Table 5.1, we can claim that the

$B_{i,j}$	Constant (M_1)	Exponential (M_2)	Migration (M_3)
Constant (M_1)	-	∞	0.92
Exponential (M_2)	0.00	-	0.00
Migration (M_3)	1.08	∞	-

Table 5.2: Bayes Factors for the Constant Model data analysis.

evidence in favour of the data having been generated under the Constant Model compared to the Migration Model is *not worth mentioning*. Obviously this is not the case, and so for our Constant Model data, we obtain a misleading result using Bayes Factors (although, correctly, we do strongly reject the Exponential Model).

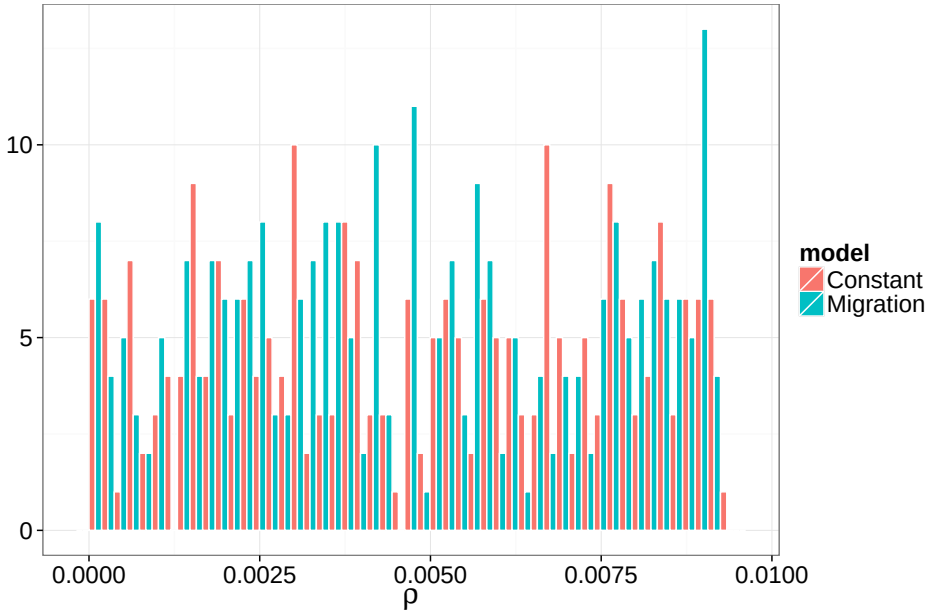


Figure 5.1: Frequency Histograms of the posterior distances for the Constant Model analysis.

If we allow the number of retained simulations to range from 50 to 2000 (for the posterior sample), we find no significant change in the proportion of posterior simulations for each model (see Figure 5.2).

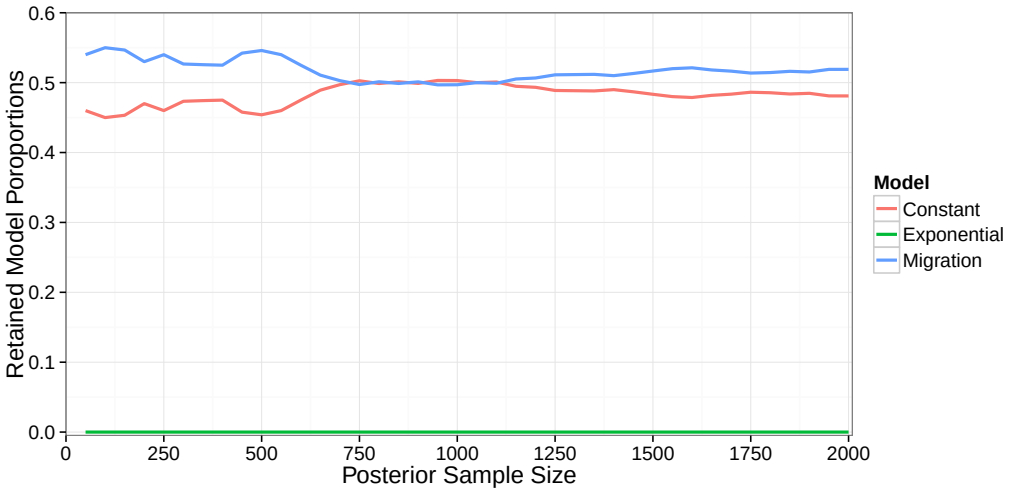


Figure 5.2: The proportion of posterior samples retained of each model for the Constant ObsDat for varying posterior sample sizes.

5.2.2 Bayes Factors for the Exponential Model Data

For the Exponential Model ObsDat, we produce a table of Bayes Factors (see Table 5.3), and a histogram of the distances, ρ, for the 500 closest retained samples (see Figure 5.3).

Of the 500 closest simulated data sets, 258 were generated under the Constant Model, 0 were generated under the Exponential Model, and 242 were generated under the Migration Model.

$B_{i,j}$	Constant (M_1)	Exponential (M_2)	Migration (M_3)
Constant (M_1)	-	∞	1.07
Exponential (M_2)	0.00	-	0.00
Migration (M_3)	0.94	∞	–

Table 5.3: Bayes Factors for the Exponential data analysis.

From Table 5.2 we observe that the Constant Model and the Migration Model are equally likely models, but the Exponential Model is infinitely less likely than either

of these models, given our data. Again the evidence in favour of the data having been generated under the Constant Model compared to the Migration Model is *not worth mentioning*. Given that we know that the data was simulated under an Exponential Model, our Bayes Factors have returned an incorrect result, as we strongly reject the correct model.

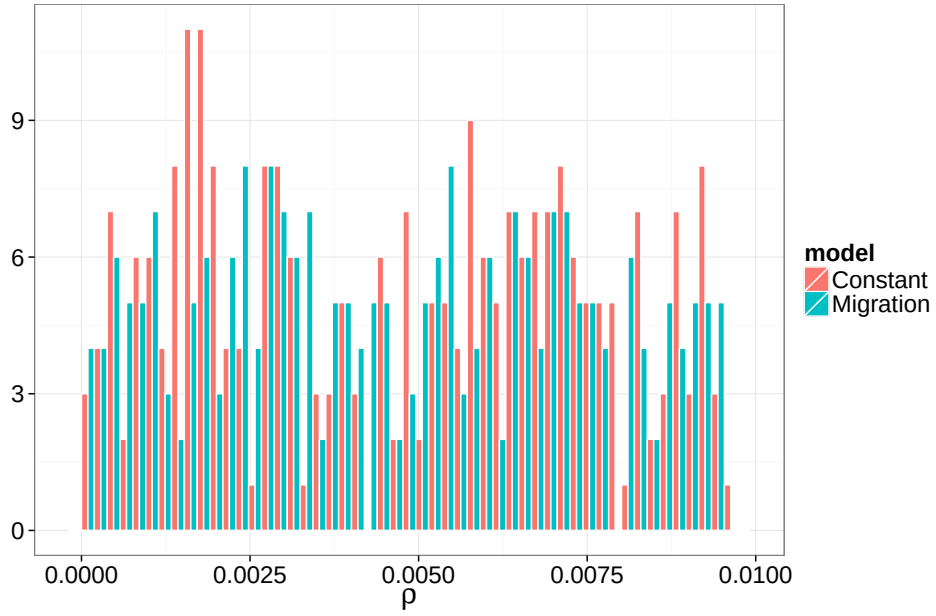


Figure 5.3: Frequency Histograms of the posterior distances for the Exponential Model analysis.

If we allow the number of retained simulations to range from 50 to 2000 (for the posterior sample), we find no significant change in the proportion of posterior simulations for each model (see Figure 5.4).

5.2.3 Bayes Factors for the Migration Model Data

For the Migration Model ObsDat, we produce a table of Bayes Factors (see Table 5.4), and a histogram of the distances, ρ, for the 500 closest retained samples (see Figure 5.5).

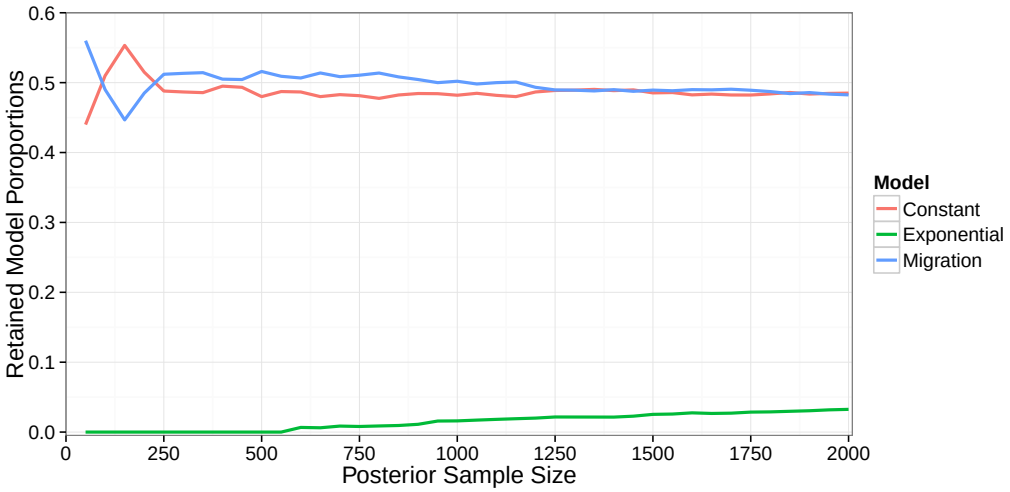


Figure 5.4: The proportion of posterior samples for each models for the Exponential ObsDat.

Of the 500 closest simulated data sets, 245 were generated under the Constant Model, 1 was generated under the Exponential Model, and 254 were generated under the Migration Model.

$B_{i,j}$	Constant (M_1)	Exponential (M_2)	Migration (M_3)
Constant (M_1)	-	735.00	0.96
Exponential (M_2)	0	-	0
Migration (M_3)	1.04	762.00	-

Table 5.4: Bayes Factors for the Migration data analysis.

From Table 5.4 we observe that the Constant model is 735 times more likely than the Exponential Model, and that the Migration Model is 762 times more likely than the Exponential Model, given the data. Again we have very strong evidence that the data was not generated under the Exponential Model. However, the evidence in favour of the data having been generated under the Migration Model compared to the Constant Model is *not worth mentioning*. Like for the Constant Model

ObsDat, we have rejected one of the incorrect models, but can not differentiate between the correct model, and the remaining incorrect model. Again the Bayes Factor has returned a rather unhelpful result.

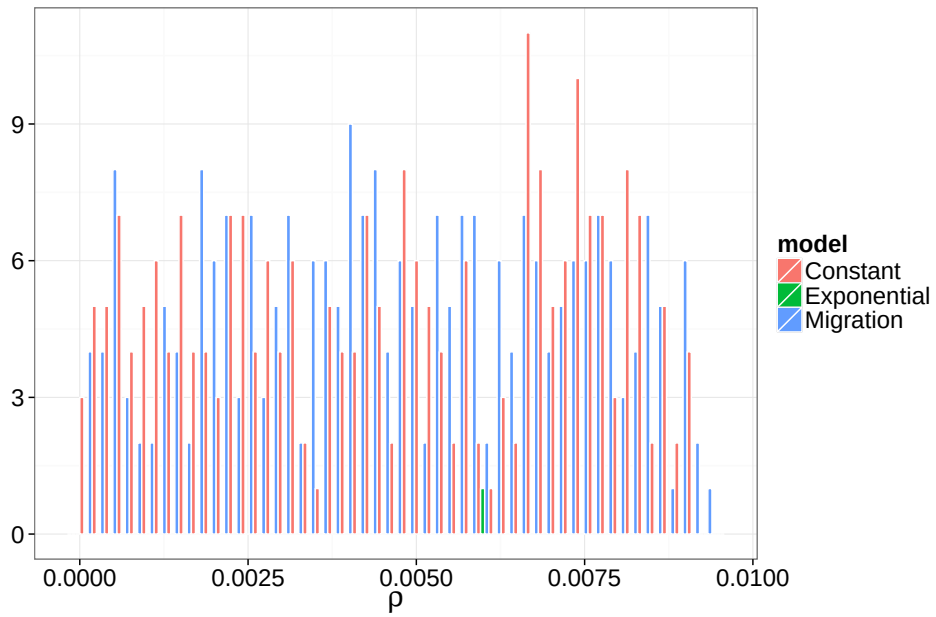


Figure 5.5: Frequency Histograms of the posterior distances for the Migration Model analysis.

If we allow the number of retained simulations to range from 50 to 2000 (for the posterior sample), we find no significant change in the proportion of posterior simulations for each model (see Figure 5.6).

5.3 ABC for model selection

In Section 5.1.1 we discussed common, but 'avoidable', problems inherent when employing model comparison in ABC schemes. These topics are well understood, and hence should be well accounted for in analyses.

A more serious, and somewhat fundamental problem exists even when we are working with *sufficient* summary statistics. We discuss this issue before discussing

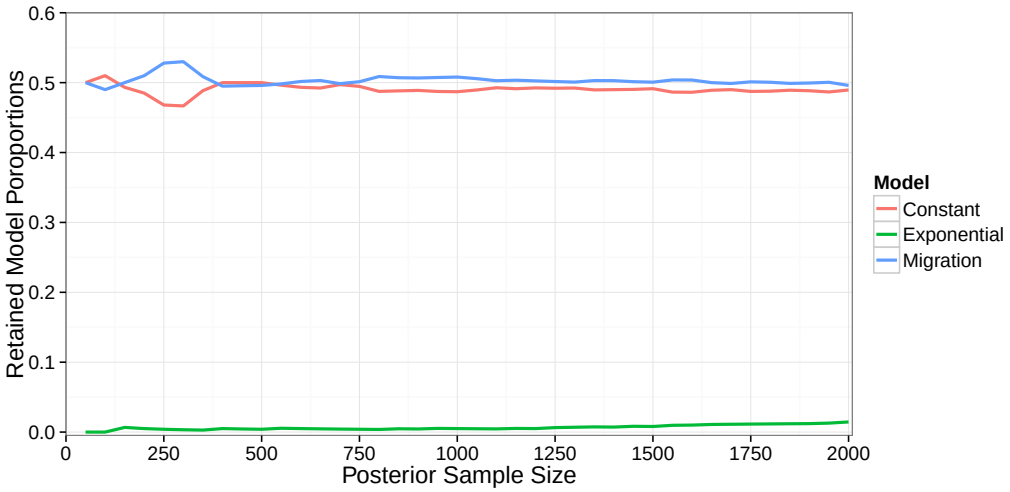


Figure 5.6: The proportion of posterior samples for each models for the Migration ObsDat.

some of the unique issues with our own methodology.

5.3.1 A Fundamental Model Comparison Problem

We present the discussion as given by Roberts *et al.* [**?**], .

Consider a set of candidate models $\mathcal{M} = \{M_1, \ldots, M_q\}$, with associated parameter vectors $\mathbf{\Theta_i} \in \Omega_i$, where $i = 1, \cdots, q$.

For the inference on this joint space of model and parameters $\{\mathcal{M}, \mathbf{\Theta_1}, \ldots, \mathbf{\Theta_q}\}$, we consider

$$P(M_i \,|\mathbf{x}) = \frac{\int_{\Omega_i} f(\mathbf{x} \,|\mathbf{\Theta_i}) r(\mathbf{\Theta_i}) d\mathbf{\Theta_i} R(M_i)}{\sum_{j=1}^{q} \int_{\Omega_j} f(\mathbf{x} \,|\mathbf{\Theta_j}) r(\mathbf{\Theta_j}) d\mathbf{\Theta_j} R(M_j)}.$$

We can replace the likelihood evaluation with a comparison of the real data $\mathbf{x}$ to

some simulated data $\mathbf{y}$ to obtain an approximate model likelihood,

$$\begin{aligned} P(M_i \,|\mathbf{x}) &\approx \hat{p}(M_i \,|\mathbf{x}) \\ &= \frac{\int_{\Omega_i}\int_{\mathbb{R}^d} I(\mathbf{x},\mathbf{y}) f(\mathbf{y}\,|\boldsymbol{\Theta}_\mathbf{i}) r(\boldsymbol{\Theta}_\mathbf{i}) d\boldsymbol{\Theta}_\mathbf{i} d\mathbf{y} R(M_i)}{\sum_{j=1}^{q}\int_{\Omega_j}\int_{\mathbb{R}^d} I(\mathbf{x},\mathbf{y}) f(\mathbf{y}\,|\boldsymbol{\Theta}_\mathbf{j}) r(\boldsymbol{\Theta}_\mathbf{j}) d\boldsymbol{\Theta}_\mathbf{j} d\mathbf{y} R(M_j)}, \end{aligned}$$

where

$$I(\mathbf{x},\mathbf{y}) = \begin{cases} 1, & \text{if } \rho\,(\mathbf{x},\mathbf{y}) \leq \epsilon, \\ 0, & \text{else.} \end{cases}$$

Clearly, $p(M_i \,|\mathbf{x}) = \hat{p}(M_i \,|\mathbf{x})$ when $\epsilon = 0$.

If we replace a comparison of $\mathbf{x}$ and $\mathbf{y}$ with a comparison of the corresponding summary statistics $S(\mathbf{x})$ and $S(\mathbf{y})$, we find the ABC model likelihood does not necessarily approximate the true model likelihood, given the data. Observe,

$$\hat{p}(M_i \,|S(\mathbf{x})\,) = \frac{\int_{\Omega_i}\int_{\mathbb{R}^w} \hat{I}(S(\mathbf{y})) g_{\boldsymbol{\Theta}_\mathbf{i}}(S(\mathbf{y})\,|\boldsymbol{\Theta}_\mathbf{i}) r(\boldsymbol{\Theta}_\mathbf{i}) d\boldsymbol{\Theta}_\mathbf{i} dS(\mathbf{y}) R(M_i)}{\sum_{j=1}^{q}\int_{\Omega_j}\int_{\mathbb{R}^w} \hat{I}(S(\mathbf{y})) g_{\boldsymbol{\Theta}_\mathbf{j}}(S(\mathbf{y})\,|\boldsymbol{\Theta}_\mathbf{j}) r(\boldsymbol{\Theta}_\mathbf{j}) d\boldsymbol{\Theta}_\mathbf{j} dS(\mathbf{y}) R(M_j)}, \quad (5.1)$$

where

$$\hat{I}(\mathbf{x},\mathbf{y}) = \begin{cases} 1, & \text{if } \rho\,(S(\mathbf{x}),S(\mathbf{y})) \leq \epsilon, \\ 0, & \text{else.} \end{cases}$$

An exact likelihood for the model M_i given the summary statistic $S(\mathbf{x})$ can be obtained by using the likelihood of the summary statistic $S(\mathbf{x})$ given the parameter vector Θ_i,

$$p(M_i \,|S(\mathbf{x})\,) = \frac{\int_{\Omega_i} g_{\Theta_i}(S(\mathbf{x})\,|\boldsymbol{\Theta}_\mathbf{i}) r(\boldsymbol{\Theta}_\mathbf{i}) d\boldsymbol{\Theta}_\mathbf{i} R(M_i)}{\sum_{j=1}^{q}\int_{\Omega_j} g_{\Theta_j}(S(\mathbf{x})\,|\boldsymbol{\Theta}_\mathbf{j}) r(\boldsymbol{\Theta}_\mathbf{j}) d\boldsymbol{\Theta}_\mathbf{j} R(M_j)} \quad (5.2)$$

Recall that if $S(\mathbf{x})$ is a sufficient statistic for $\boldsymbol{\Theta}_\mathbf{i}$, then by Neymann-Pearson factorisation we have,

$$\begin{aligned} f(\mathbf{x}\,|\boldsymbol{\Theta}_\mathbf{i}) &= h_i(\mathbf{x}\,|S(\mathbf{x})) g_{\boldsymbol{\Theta}_\mathbf{i}}(S(\mathbf{x})\,|\boldsymbol{\Theta}_\mathbf{i}) \\ \implies g_{\boldsymbol{\Theta}_\mathbf{i}}(S(\mathbf{x})\,|\boldsymbol{\Theta}_\mathbf{i}) &= \frac{f(\mathbf{x}\,|\boldsymbol{\Theta}_\mathbf{i})}{h_i(\mathbf{x}\,|S(\mathbf{x}))}, \end{aligned} \quad (5.3)$$

for model M_i.

By substituting equation (5.3) into equation (5.2), we obtain

$$\begin{aligned} p(M_i \,|S(\mathbf{x})) &= \frac{\int_{\Omega_i} \frac{f(\mathbf{x}|\boldsymbol{\Theta}_\mathbf{i})}{h_i(\mathbf{x}|S(\mathbf{x}))} r(\boldsymbol{\Theta}_\mathbf{i}) d\boldsymbol{\Theta}_\mathbf{i} R(M_i)}{\sum_{j=1}^{q} \int_{\Omega_j} \frac{f(\mathbf{x}|\boldsymbol{\Theta}_\mathbf{i})}{h_j(\mathbf{x}|S(\mathbf{x}))} r(\boldsymbol{\Theta}_\mathbf{j}) d\boldsymbol{\Theta}_\mathbf{j} R(M_j)} \\ &= \frac{\int_{\Omega_i} f(\mathbf{x}\,|\boldsymbol{\Theta}_\mathbf{i}) r(\boldsymbol{\Theta}_\mathbf{i}) d\boldsymbol{\Theta}_\mathbf{i} R(M_i)}{h_i(\mathbf{x}\,|S(\mathbf{x}))\sum_{j=1}^{q} \frac{1}{h_j(\mathbf{x}|S(\mathbf{x}))} \int_{\Omega_j} f(\mathbf{x}\,|\boldsymbol{\Theta}_\mathbf{i}) r(\boldsymbol{\Theta}_\mathbf{j}) d\boldsymbol{\Theta}_\mathbf{j} R(M_j)} \\ &\neq p(M_i \,|\mathbf{x}), \end{aligned}$$

unless $h_i(\mathbf{x}\,|S(\mathbf{x})) = h_j(\mathbf{x}\,|S(\mathbf{x}))$ for all $i, j = 1, \ldots, q$.

That is, the likelihood of the model M_i, given the data $\mathbf{x}$ is not necessarily the same as the likelihood of the model, given the associated summary statistic, unless the probability of seeing the data given the summary statistic is the same for each model. Hence, the approximate model likelihood $\hat{p}(M_i \,|S(\mathbf{x}))$ given in equation (5.1), will not approximate the model likelihood $P(Y = M_i \,|\mathbf{x})$, even if $\epsilon = 0$.

By taking the ratio of $p(M_i \,|\mathbf{x})$ and $p(M_j \,|\mathbf{x})$, we remove some of the possible bias introduced by the $h_j(\mathbf{x}\,|S(\mathbf{x}))$ terms. Note,

$$\frac{p(M_i \,|\mathbf{x})}{p(M_j \,|\mathbf{x})} = \frac{\int_{\Omega_i} \frac{1}{h_i(\mathbf{x}|S(\mathbf{x}))} f(\mathbf{x}\,|\boldsymbol{\Theta}_\mathbf{i}) r(\boldsymbol{\Theta}_\mathbf{i}) d\boldsymbol{\Theta}_\mathbf{i} R(M_i)}{\int_{\Omega_j} \frac{1}{h_j(\mathbf{x}|S(\mathbf{x}))} f(\mathbf{x}\,|\boldsymbol{\Theta}_\mathbf{j}) r(\boldsymbol{\Theta}_\mathbf{j}) d\boldsymbol{\Theta}_\mathbf{j} R(M_j)},$$

and hence we are still left with a bias of

$$\frac{h_j(\mathbf{x}\,|S(\mathbf{x}))}{h_i(\mathbf{x}\,|S(\mathbf{x}))}$$

for our estimate of B_{ij}.

5.3.2 Further ABC Model Comparison Issues

In addition to the common problems associated with model inference in ABC, and the more fundamental issue outlined above, we suffer from two final problems when comparing our posterior ratios.

First, each of the models produce transformed estimates of the posterior initial effective population size, under different transformations. Clearly then, we must back-transform these posterior estimates if we are to compare distances under different models, and this has been done in the Bayes Factor analyses presented in Section 5.1.1.

Note also, that our back-transformed posterior mean estimates for the initial effective population sizes had sample standard deviations 40306.42, 38267.47 and 10163.74 for the Constant, Exponential and Migration Models respectively. The posterior mean estimate for the growth rate K in the Exponential model had sample standard deviation 2.159×10^{-5}. So an increase in the estimated posterior mean initial population size of one standard deviation corresponds to an increase in 2.7798×10^{9} standard deviations in the estimated posterior growth rate (with respect to Euclidean distance).

To account for the discrepancy in the weighted deviation from estimated means for both parameters in the Exponential Model, all of the back-transformed posterior estimates were normalised to have a sample mean of zero, and sample standard deviation of one. What these two transformations (the power back transformation and the normalisation) have done to our comparisons of the distances for each model is not obvious.

Consider comparing a distance in a one-dimensional space, with a distance in a two-dimensional space.

For the Constant and Migration Models, we calculate the distance of a given set of predicted posterior means $\hat{\Phi}$ from our observed predicted posterior means $\hat{\Phi}^{obs}$, which we denote

$$\rho_1(\hat{\Phi}, \hat{\Phi}^{obs}) = \sqrt{\left(\hat{N}_0 - \hat{N}_0^{obs}\right)^2}. \tag{5.4}$$

For the Exponential Model we use

$$\rho_2(\hat{\Phi}, \hat{\Phi}^{obs}) = \sqrt{\left(\hat{N}_0 - \hat{N}_0^{obs}\right)^2 + \left(\hat{K} - \hat{K}^{obs}\right)^2}. \tag{5.5}$$

For the data generated under the Exponential Model, the back-transformed, normalised distances for each parameter, and each candidate model, are given in Figure 5.7. We can see that the shape of the posterior distribution for the initial population size is reasonably similar for each population model, and that each is centred somewhere above zero. For the Constant and Migration Models, the difference between the ABCDat posterior mean estimates from the ObsDat estimated initial population size are not centred around zero. These two distributions have a mean of 1.472 and 1.267 respectively, with 91.41% and 86.02% of the observed differences being greater than 0.

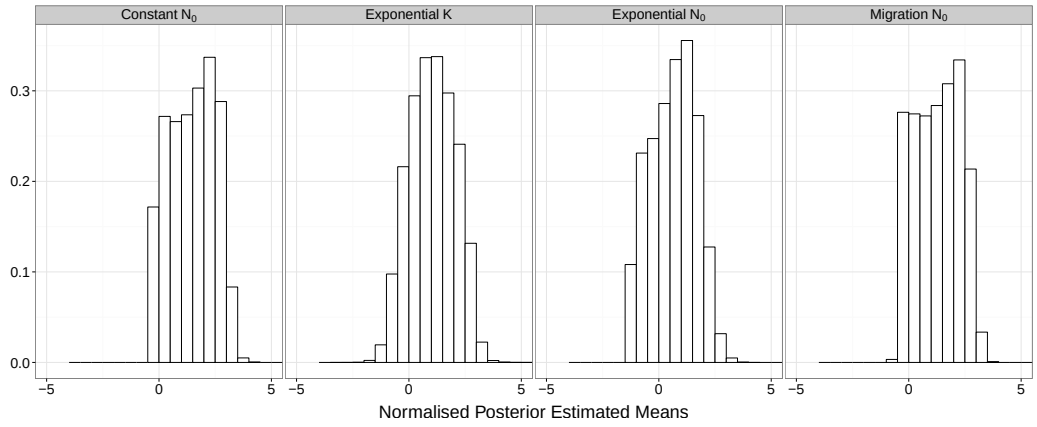


Figure 5.7: The normalised, back-transformed posterior mean estimates.

For the Exponential Model though, the difference between the ABCDat posterior mean estimates from the ObsDat estimated initial population size is 0.6352, which is relatively closer than for the Constant and Migration Models (recall we have a common sample standard deviation of one). Similarly, only 70.67% of the observed differences are greater than zero. What we observe when we look at this is that our posterior differences for the initial population size estimates are more closely centred about zero for the Exponential model. That is, our guesses are 'closer' for the Exponential Model on average, when only considering N_0.

However, when we use the Euclidean Distance from equation (5.5), we also take into account how far our back-transformed, normalised estimates of the posterior

mean growth rate K are from our estimates from our observed data. The differences for the estimated posterior mean growth rate from the estimate obtained from the ObsDat have a sample mean of 1.051, and 83.21% of these distances are greater than zero. While these differences have a more symmetric distribution, we will still add some non-zero amount to the distance when accounting for the difference in growth rate estimates.

That is, on average

$$\rho_1(\hat{\Phi}, \hat{\Phi}^{obs}) \leq \rho_2(\hat{\Phi}, \hat{\Phi}^{obs}),$$

since although the initial effective population size estimates are closer for the Exponential model, on average, the combined effect of the initial effective population size and the growth rate is not. Since this is the case for the Exponential Model ObsDat, the pronounced effect of this two-dimensional distance is only made worse in the case of the Constant and Migration Model ObsDat (see Figure 5.8).

We propose the use of a different distance metric $\rho_C(\hat{\Phi}, \hat{\Phi}^{obs})$ that is not *as sensitive* to differences in dimensionality between parameter spaces called the Chebychev Distance [7]. This is defined as

$$\rho_C(\hat{\Phi}, \hat{\Phi}^{obs}) = \max_{k=1}^{P} \left| \hat{\phi}_k - \hat{\phi}_k^{(obs)} \right|.$$

This distance metric will only take into account the 'worst' case estimate (of the growth rate and the initial effective population size). Since

$$\max\left(\left| \hat{N}_0 - \hat{N}_0^{obs} \right|, \left| \hat{K} - \hat{K}^{obs} \right| \right) \leq \sqrt{\left(\hat{N}_0 - \hat{N}_0^{obs} \right)^2 + \left(\hat{K} - \hat{K}^{obs} \right)^2},$$

by the triangle inequality,

$$\rho_C(\hat{\Phi}, \hat{\Phi}^{obs}) \leq \rho_2(\hat{\Phi}, \hat{\Phi}^{obs}).$$

That is, our observed distances under the Chebychev distance will be less than or equal to those observed under the Euclidean distance. However, when recalculating

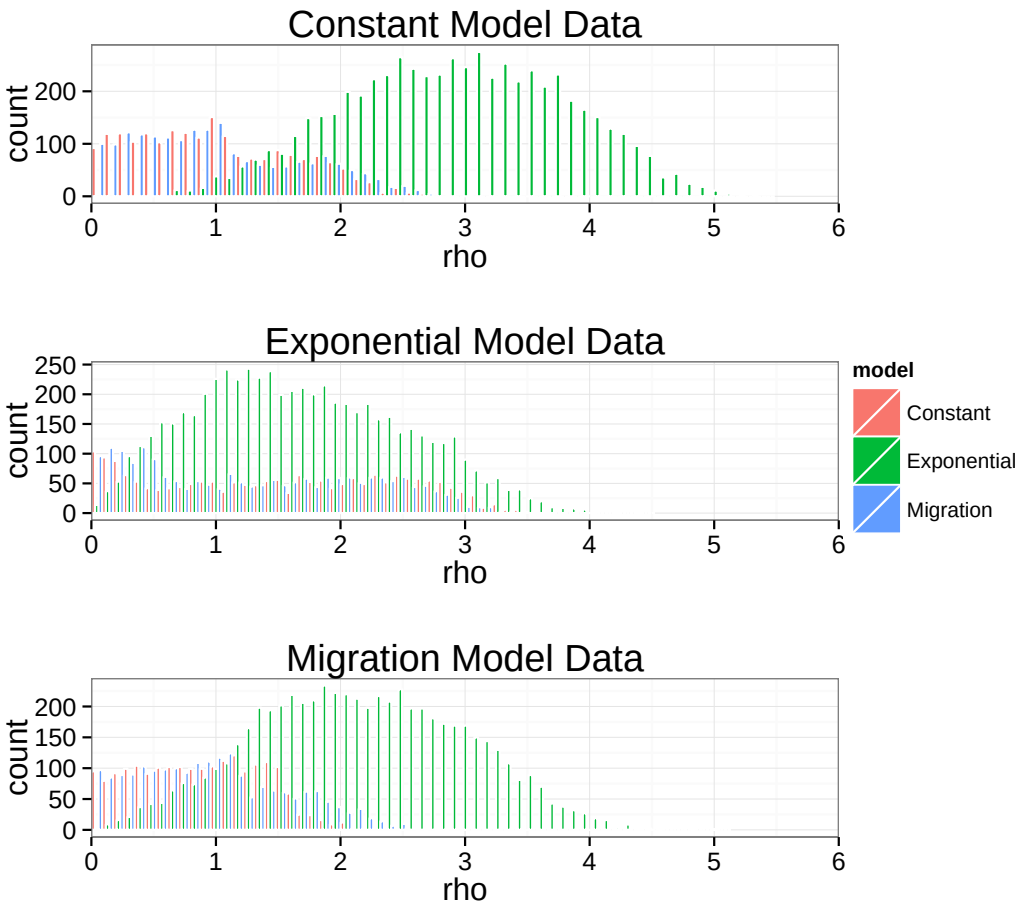


Figure 5.8: Distances for each model in the back-transformed space (note there are three times as many Exponential Model simulations).

the Bayes Factors for the Exponential Obsdat, although the Exponential Model distances were reduced, the Bayes Factors did not change.

An initial consideration is that we explored the one-dimensional support of the prior distribution for the initial population size for the Constant and Migration Models, with 50000 simulations. In comparison, we explored the two-dimensional support of the posterior distributions of the initial population size and growth rates with 150000 simulations. Ideally, if we explore a one-dimensional support with d simulations, we would like to explore a two-dimensional support with d^2 simulations if we wish to compare our results. This will ensure the posterior distri-

butions (if they are even comparable in 'how much they differ from the posterior distributions') are equally well explored. This means that for the 50000 simulations for the Constant and Migration Models, we would have required 2.5×10^9 simulations for the Exponential Model. This is beyond the computational scope of this project.

While it is infeasible to increase the number of simulated observations for the Exponential Model to the required amount, we can *reduce* the number of simulations for the Constant and Migration Models. Arguably then, if our Bayes Factors' poor performance is a result of 'under exploring' the parameter space for the Exponential Model, as we reduce the number of simulations for the Constant and Migration Models, the associated Bayes Factor should tend towards evidence for the Exponential Model, in the case of the Exponential ObsDat.

In Figure 5.9 we consider prior simulation sample sizes for the Constant and Migration Models of 5000 samples, up to the complete 50000 samples, in steps of 100. For each reduced sample size, we took five random subsamples of our full prior samples, calculated the Bayes Factor for each subsample, and recorded the mean Bayes Factor.

Clearly the Bayes factors B_{21} and B_{23} do not tend to some value of strong evidence in favour of the Exponential Model. A maximum value of $B_{21} = 3.3 \times 10^{-4}$ is achieved, and this is still considered 'Very Strong' evidence in favour of the Constant Model, over the Exponential Model. It seems then that the amount we have explored our parameter spaces was satisfactory for both the one-dimensional and two-dimensional spaces.

A second consideration is that the prior distributions were not comparable. That is, if the prior distribution for a parameter is centred tightly around the true parameter value for one of our models, we would expect that this model will produce more simulations like the true data, and hence be retained more often. Similarly, if the correct model had a parameter with a particularly poor prior

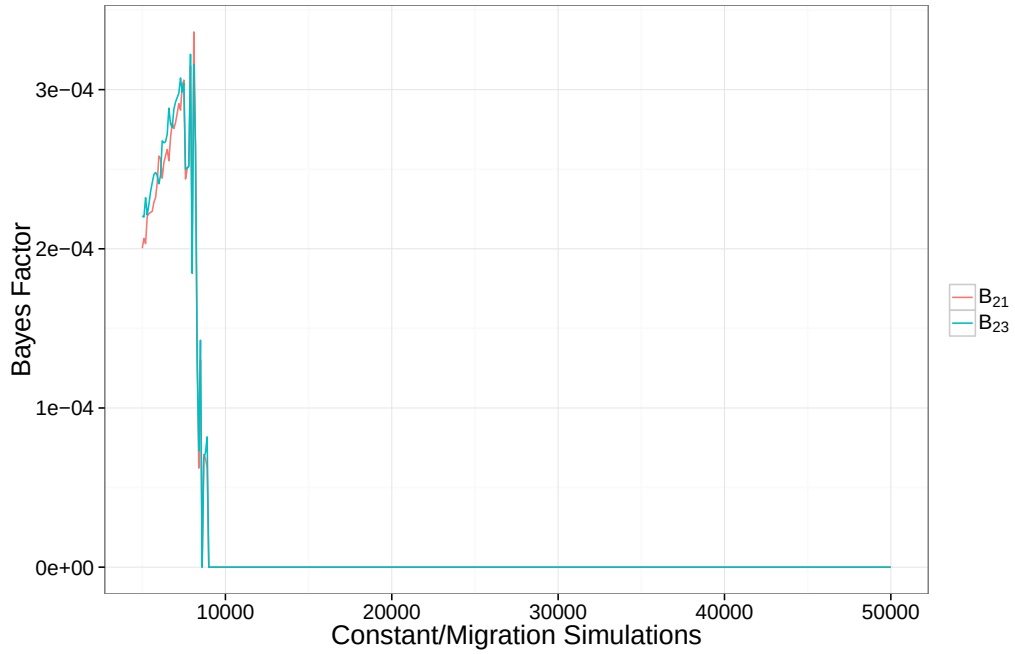


Figure 5.9: Bayes Factors for smaller prior samples for the Constant and Migration Models for the Exponential ObsDat.

distribution (with the true parameter value somewhere in the tail), then many simulations with large distances will be produced.

The prior distributions for the initial effective population sizes seemed comparable for each population model (in that their posterior distributions appear similar). The prior distribution for the growth rate may not be similarly comparable, and hence the combined effect of both the prior distributions for the growth rate and the initial effective population size for the Exponential Model may make the resultant Chebychev distances substantially incomparable.

To investigate this, we restricted the prior distribution for the growth rate K to be between $\ln(0.15)/2000$ and $\ln(0.35)/2000$ (recall, previously K fell between $\ln(0.15)/2000$ and $\ln(0.75)/2000$). This reduced our number of simulations for the Exponential Model to be of size 50191, hence we also reduced the number of simulations for the Constant and Migration Models to be of size 5000 to avoid over

exploring the one-dimensional parameter spaces.

The new Bayes Factor analyses yielded two differences. The Bayes Factor for the Exponential Obsdat was *positive* in support of the Exponential Model over the Constant and Migration Models with Bayes Factors of $B_{21} = 18.17$ and $B_{23} = 10.09$ respectively. So the Bayes Factor is now returning a correct result for the Exponential ObsDat.

For the Migration ObsDat, we now produce a Bayes Factor $B_{23} = 25.02$. That is, we have very *strong* evidence that the Migration ObsDat was produced under the Exponential Model when compared to the Migration Model. So this change in prior distribution for the growth rate has resulted in a correct result for the Exponential ObsDat, but has returned a completely incorrect result for the Migration ObsDat (there was no change in Bayes Factor for the Constant Model).

In trying to correct for the many types of error we have suggested, we have never been able to return a correct result for all three Bayes Factor analyses. It seems that in the case where we have introduced transformations to our posterior sample (power transformation and normalisation), Bayes Factors simply do not perform well. However, even without the introduction of transformations, we have also shown that Bayes Factor suffers from a fundamental bias. This bias is a result of the probability of the data, given the observed summary statistics, not being equal for all models.

In the remainder of this chapter, we suggest a method of data classification that lends itself naturally to a simulated data environment and the already defined summary statistics, insufficient as they may be.

5.4 Multinomial Logistic Regression (MLR) for Model Selection.

Our method of parameter estimation from Chapter 4 is an example of 'supervised learning' in that we have derived a function for parameter estimation from a labelled training data set [26]. We chose this method due to the ease with which we can produce simulations, and the intractability of the associated likelihood functions. It is natural to extend this idea of supervised learning to model classification.

We wish to define a function that takes an observed data set, and returns the most likely model that the data was produced under (with an associated probability), from a set of candidate models. We define our function using a 'training set' of data in the following way.

Let $\mathbf{X}$ be the observed data, where $\mathbf{x}^k$ is the k^{th} row of $\mathbf{X}$. For J categorical outcomes we (arbitrarily) choose category J as the base case for comparison, and define the model

$$\ln\left(\frac{P\left(Y^k = c\,|\mathbf{X}\right)}{P\left(Y^k = J\,|\mathbf{X}\right)}\right) = \boldsymbol{\beta}^c \cdot \mathbf{x}^k \tag{5.6}$$

where Y^k is the category of the k^{th} observation, $\boldsymbol{\beta}^c = (\beta_0^c, \beta_1^c, \ldots, \beta_P^c)$, $c = 1, \ldots, J-1$ and $\mathbf{x}^k = (x_0^k, x_1^k, \ldots, x_P^k)$. (It is important to note that while we know the category of the k^{th} observation, we aim to find the values for each $\boldsymbol{\beta}^c$ such that our model 'fits the data best').

From equation (5.6) we observe that

$$\begin{aligned} \ln\left(\frac{P\left(Y^k = J\,|\mathbf{X}\right)}{P\left(Y^k = J\,|\mathbf{X}\right)}\right) &= \boldsymbol{\beta}^J \cdot \mathbf{x}^k \\ \implies \ln(1) &= \boldsymbol{\beta}^J \cdot \mathbf{x}^k \\ \implies 0 &= \boldsymbol{\beta}^J \cdot \mathbf{x}^k \\ \implies \boldsymbol{\beta}^J &= \mathbf{0}, \end{aligned} \tag{5.7}$$

assuming $\mathbf{x}^k \neq \mathbf{0}$.

Again using equation (5.6), we find

$$\begin{aligned} \frac{P\left(Y^k = c\,|\mathbf{X}\right)}{P\left(Y^k = J\,|\mathbf{X}\right)} &= e^{\boldsymbol{\beta}^c \cdot \mathbf{x}^k} \\ \implies P\left(Y^k = c\,|\mathbf{X}\right) &= P\left(Y^k = J\,|\mathbf{X}\right) e^{\boldsymbol{\beta}^c \cdot \mathbf{x}^k}, \end{aligned}$$

for $c = 1, \ldots, J-1$.

By the law of total probability we know that

$$\begin{aligned} &\sum_{c=1}^{J} P\left(Y^k = c\,|\mathbf{X}\right) = 1 \\ \implies &\sum_{c=1}^{J-1} P\left(Y^k = c\,|\mathbf{X}\right) + \sum_{c=1}^{J} P\left(Y^k = J\,|\mathbf{X}\right) = 1 \\ \implies &P\left(Y^k = J\,|\mathbf{X}\right) \sum_{c=1}^{J-1} e^{\boldsymbol{\beta}^c \cdot \mathbf{x}^k} + \sum_{c=1}^{J} P\left(Y^k = J\,|\mathbf{X}\right) = 1 \\ \implies &P\left(Y^k = J\,|\mathbf{X}\right) = \frac{1}{1 + \sum_{c=1}^{J-1} e^{\boldsymbol{\beta}^c \cdot \mathbf{x}^k}}, \end{aligned}$$

and hence

$$P\left(Y^k = i\,|\mathbf{X}\right) = \frac{e^{\boldsymbol{\beta}^i \cdot \mathbf{x}^k}}{1 + \sum_{c=1}^{J-1} e^{\boldsymbol{\beta}^c \cdot \mathbf{x}^k}}. \tag{5.8}$$

From (5.7) we know that

$$\boldsymbol{\beta}^J = \mathbf{0},$$

and so we can rewrite equation (5.8) as

$$P\left(Y^k = i\,|\mathbf{X}\right) = \frac{e^{\boldsymbol{\beta}^i \cdot \mathbf{x}^k}}{\sum_{c=1}^{J} e^{\boldsymbol{\beta}^c \cdot \mathbf{x}^k}}, \quad i = 1, 2, \cdots, J. \tag{5.9}$$

We now select the values for $\boldsymbol{\beta} = \left(\boldsymbol{\beta}^1, \boldsymbol{\beta}^2, \ldots, \boldsymbol{\beta}^{J-1}, \mathbf{0}\right)$ that 'best fit' our model classification training data via Maximum Likelihood using the log-likelihood of $\boldsymbol{\beta}$,

$$\begin{aligned}\ell\left(\boldsymbol{\beta}\,|\mathbf{X}\right) &= \sum_{k=1}^{n}\sum_{j=1}^{J} \ln\left(P\left(Y^k = i\,|\mathbf{X}\right)\right) \\ &= \sum_{k=1}^{n}\sum_{j=1}^{J} \ln\left(\frac{e^{\boldsymbol{\beta}^i\cdot\mathbf{x}^k}}{\sum_{c=1}^{J} e^{\boldsymbol{\beta}^c\cdot\mathbf{x}^k}}\right) \quad \text{from (5.9).}\end{aligned}$$

A Fisher-Scoring iterative estimation method is then applied to a chosen starting point for $\boldsymbol{\beta}$, $\boldsymbol{\beta}^{(0)}$, and we update $\boldsymbol{\beta}^{(\nu)}$ iteratively via,

$$\boldsymbol{\beta}^{(\nu+1)} = \boldsymbol{\beta}^{(\nu)} + \alpha\left(\frac{\partial^2\ell(\boldsymbol{\beta}\,|\mathbf{X})}{\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}}\bigg|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(\nu)}}\right)^{-1}\frac{\partial\ell\left(\boldsymbol{\beta}\,|\mathbf{X}\right)}{\partial\boldsymbol{\beta}}\bigg|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(\nu)}}$$

until 'convergence'. Convergence is met when any of the three following conditions are satisfied;

1. $\left|\ell\left(\boldsymbol{\beta}^{(\nu+1)}\,|\mathbf{X}\right) - \ell\left(\boldsymbol{\beta}^{(\nu)}\,|\mathbf{X}\right)\right| < \epsilon_k$,
2. $\max\left|\boldsymbol{\beta}^{(\nu+1)} - \boldsymbol{\beta}^{(\nu)}\right| < \epsilon_p$,
3. $\max\frac{\partial\ell(\boldsymbol{\beta}|\mathbf{X})}{\partial\boldsymbol{\beta}}\Big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(\nu)}} < \min(\epsilon_k, \epsilon_p)$.

For an exhaustive discussion on the topic, we direct the reader to 'Generalised Linear Models' [22].

5.4.1 MLR Model Classification Results

We trained our MLR classifier with the same TrainDat (with the same prior distributions) as we did for our linear models for parameter estimation. For a Table of parameter values and prior distributions for these data, refer to Table 4.2. The analysis was performed in the `R-Statistical Software Package` using the `multinom()` function in the `nnet` package [46].

For each ObsDat (Constant, Exponential and Migration Models) in Chapter 4, for which we performed parameter estimation, we now use our trained MLR to classify the data as being the product of a Constant, Exponential or Migration Model dynamics. In Table 5.5 we return the probabilities of the data being generated under these three models (for the list of the 33 significant predictor see Section A.7).

	$P(M_C \mid \mathbf{X})$	$P(M_E \mid \mathbf{X})$	$P(M_M \mid \mathbf{X})$
$\mathbf{X} = \mathbf{X}_C$	0.99997	4.1317×10^{-5}	9.4891×10^{-3}
$\mathbf{X} = \mathbf{X}_E$	8.8063×10^{-6}	0.99995	1.7517×10^{-21}
$\mathbf{X} = \mathbf{X}_M$	1.6719×10^{-5}	7.2170×10^{-17}	0.99051

Table 5.5: Predicted model classification probabilities for M_C,M_E and M_M and $\mathbf{X}_C$,$\mathbf{X}_E$ and $\mathbf{X}_M$ are the Constant, Exponential and Migration models, and the ObsDat generated under these models, respectively.

These values are the 'soft' classifications for our data as we have not chosen which specific model we claim our Obsdat was generated under. To perform 'hard' classification, for the purpose of parameter estimation, we select the model with highest associated probability. That is, for data set $\mathbf{X}$, we select model M_j for $j \in \{1, 2, \ldots, J\}$, such that

$$j = \arg\max_i P(Y = M_i \mid \mathbf{X}).$$

Our MLR classification is hard classifying correctly for each ObsDat, and is predicting with at worst 99.05% certainty the correct modelling scheme.

To explore how well our MLR classifier performs on a large sample of simulated data, we combined our three ABCDats. Again, for a Table of parameter values and prior distributions for these data, refer to Table 4.2.

We performed a hard classification for each of the 250,000 simulations, and recorded

the proportion of times each model was selected for each type of simulation model. Our MLR classification performs extremely well, selecting the correct model for 91.5% of the Constant Model data, 99.89% of the Exponential Model data, and 97.97% of the Migration Model data (see Figure 5.10).

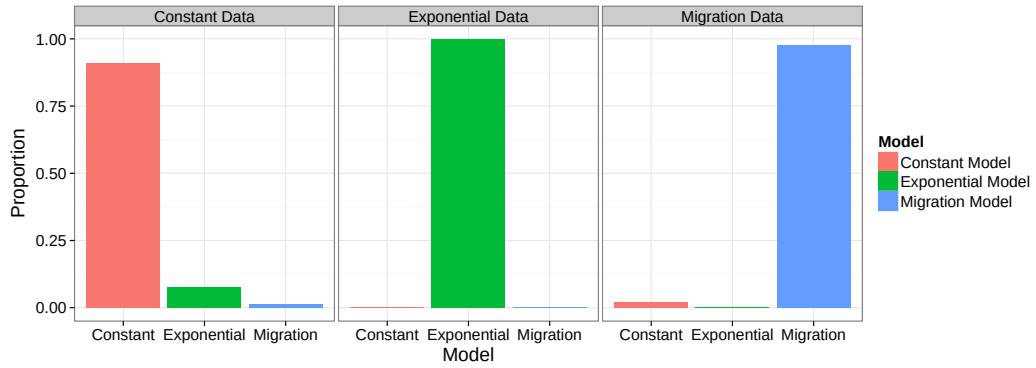


Figure 5.10: Bar charts of the proportions of model classifications for each ABCDat.

To explore why our MLR classification method predicts so well, we performed a Principal Component Analysis (PCA) on our summary statistics.

Recall, PCA is a procedure that takes a set of P observed and possibly correlated variables, and decomposes them into P orthogonal, uncorrelated variables called components. These components are linear combinations of the original variables, and are ranked such that the first component has the largest variance, and the P^{th} component has the smallest variance. In doing this, we can reduce the number of variables (and hence the dimensionality) in a data set by removing the components with negligible variance, hence retaining most of the information in our sample.

The PCA of the training data we used to define our MLR classifier decomposed our data into 42 ranked components with variances as shown in Figure 5.11. We decided that most of the variance, and hence information in the data is contained in the first two components as we observe a noticeable 'elbow' in the scree plot at the third component.

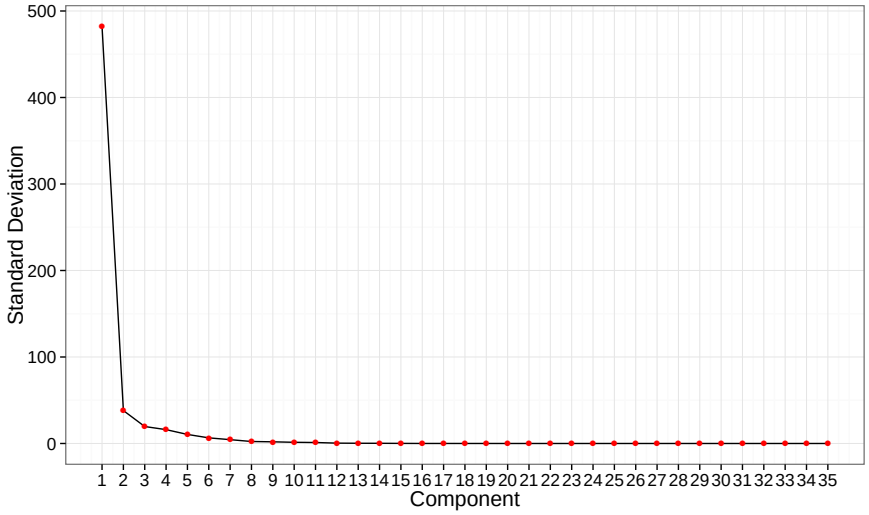


Figure 5.11: Scree plot showing the variance of the associated ranked components for our training data.

We then plotted the first two components of the combined TrainDat generated for parameter estimation, and coloured each observation by the model under which it was simulated.

The relationship between PCA and binary response logistic regression is well understood, but it is beyond the scope of this project to establish the relationship between PCA and MLR [11]. However, both MLR classification and PCA use linear combinations of the summary statistics to obtain information about our data. It makes sense that these analyses 'agree' in some way. As PCA is a dimension reduction algorithm that allows for visualisation, what we 'see' in the two-dimensional principal component should be comparable to the results of our hard MLR classification scheme.

We can see in Figure 5.12 that on the first two principal components, the Exponential Model data, and the Migration model data completely separate into two clusters. We can also see in Figure 5.10 that when the data was simulated under the Exponential Model, we never hard classified it as Migration data, and vice versa.

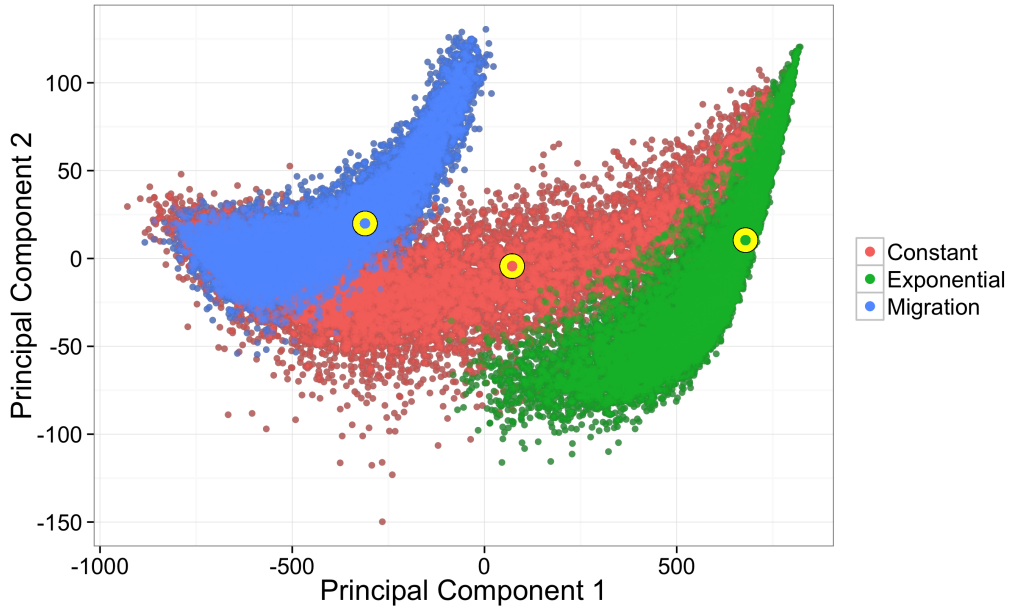


Figure 5.12: The first two principal components for our combined TrainDat data set, coloured by the model under which simulation was performed. The three yellow data points are the ObsDat.

We see some crossover between the Migration Model data and the Constant Model data in Figure 5.12, and correspondingly we misclassify the Migration Model data as Constant Model data for 2.03% of simulated observations, and the Constant Model Data as Migration Model data for 1.46% of simulations. Finally, we also see a crossover between the Constant Model data and the Exponential Model data in Figure 5.12, and hence we misclassify the Exponential Model data as Constant Model data for 0.11% of simulations, and the Constant Model data as Exponential Model data for 7.07% of simulations.

A sensitivity analysis similar to the modified growth rate K prior distribution analysis was performed for our MLR results. We again restricted the Exponential Model data set to only contain simulations where K was between $\ln(0.15)/2000$ and $\ln(0.35)/2000$, and repeated the MLR classification and principal component analyses.

When we restricted K to a tighter prior interval, the MLR classification probabilities for the ObsDat remained correctly hard classified, and marginally improved in their soft classification probabilities. Similarly, the hard classification on the data generated for parameter estimation performed well for the Constant Model and Migration Model simulations (see Figure 5.14).

The Exponential Model data clustered in a tighter two-dimensional region, and somewhat more 'separately' from the Constant Model data in that there was less crossover between the Constant Model data and the Exponential Model data (see Figure 5.13). This further separation resulted in a decrease in the Constant Model data being hard classified as Exponential Model data (down from 7.07% to 0.11% of Constant Model simulations).

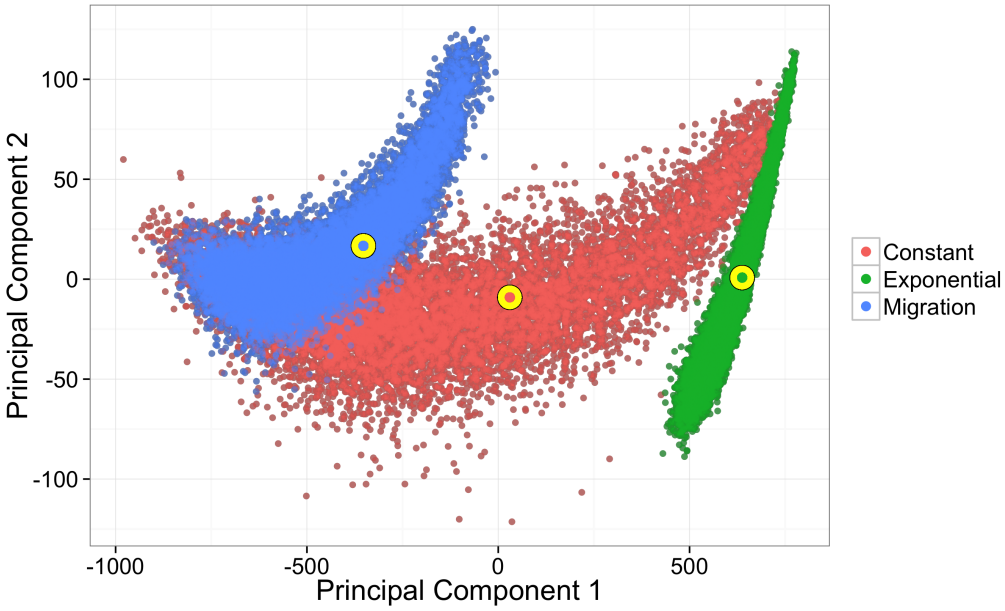


Figure 5.13: The first two principal components for our data sets (where K is between $\ln(0.15)/2000$ and $\ln(0.35)/2000$), coloured by the model under which simulation was performed. The three yellow data points are our 'real data'.

This improvement in classification for Constant Model data can probably be explained by the fact that the Constant Model is just a special case of the Expo-

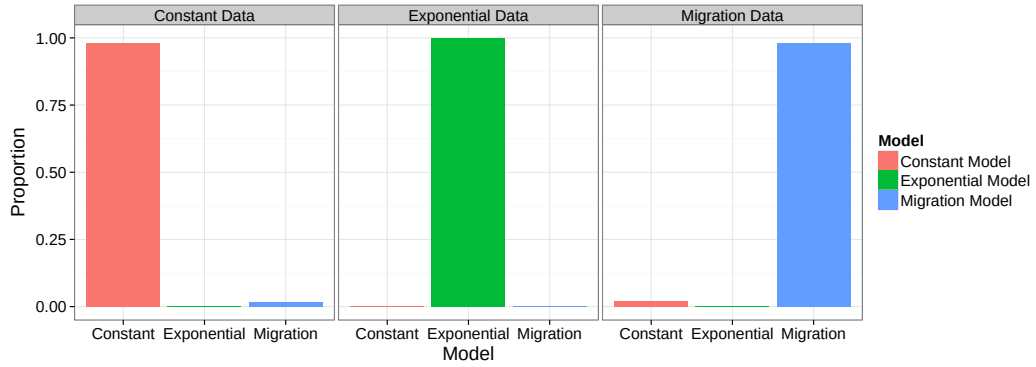


Figure 5.14: Bar charts of the proportions of model classifications for each type of data (where K is between $\ln(0.15)/2000$ and $\ln(0.35)/2000$).

nential Model where $K = 0$. By restricting K to be less than $\ln(0.35)/2000$, we enforce the effective population size to decrease proportionally to at most 35% of the initial effective population size, rather than at most 75% of the initial effective population size. The further enforced distinction between the Exponential Model and the Constant Model is reflected in the summary statistics.

5.4.2 MLR classification and Bayes Factors

Both the MLR classification scheme and our parameter estimation scheme are examples of supervised learning. A natural extension of our parameter estimation method would include a first step where MLR model classification would occur first. For our method, the inclusion of the Box-Cox transformation made a comparison of distances between models impossible to interpret. Back transformation made comparisons possible, but difficult, and we were unable to recover a correct result using the Bayes Factor analysis.

By using the MLR classification, the issue of transformation was never introduced, and we were able to correctly classify our data in each of our three analyses. However, our method also allows for an initial sensitivity analysis for the issue of the exploration of model space. Consider the Constant Model data classified

under a MLR classification scheme where we only consider the Exponential and Migration Models as elements of the candidate model set.

We retrained our MLR classification model with only Exponential and Migration Model TrainDat, and hence classification outcomes. From Figure 5.13 we see that the data clusters have further, and more distinctly separated from one another. This has resulted in the Exponential and Migration ObsDat still being hard classified correctly (this time with probability one), but the Constant ObsDat has now been classified as Exponential Model data (with probability one).

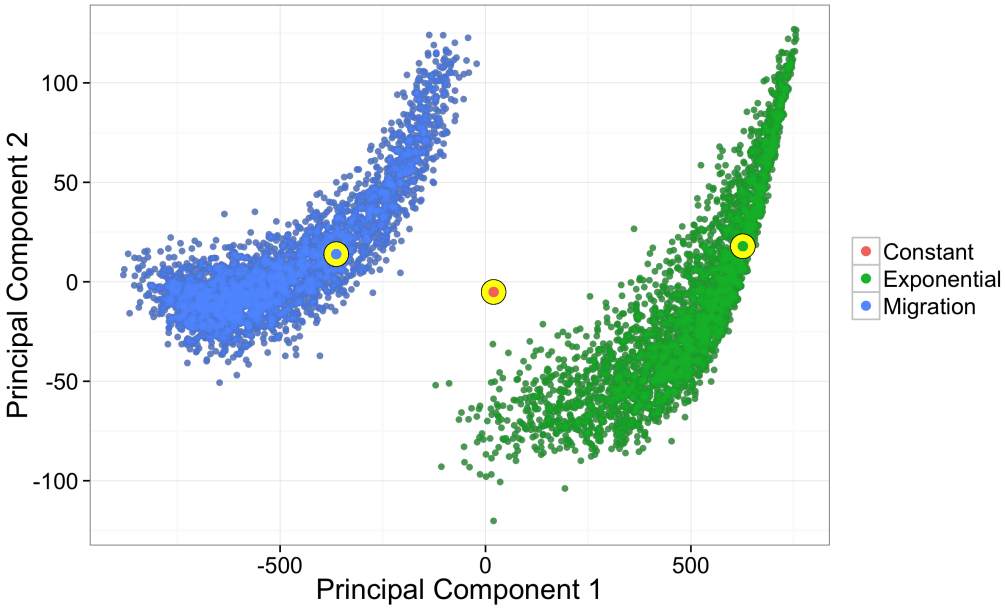


Figure 5.15: The first two principal components for the combined TrainDat data set, where the Constant Model data is removed, coloured by the model under which simulation was performed. The three yellow data points are the ObsDat.

While this misclassification of the Constant Model data is of concern, this is true of any ABC model comparison/selection method when the correct model is not included in the candidate model set. Furthermore, it is clear from Figure 5.15 that although we classify our Constant Model data as Exponential Model data, it is

not visually a member of the Exponential Model cluster, but simply closer to the Exponential data than the Migration data. We suggest a method of identifying when the MLR classification may be dubious.

Recall the centroid $\mathbf{C}$ of a set of vectors $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_k \in \mathbb{R}^n$ is defined as

$$\mathbf{C} = \frac{\mathbf{x}_1 + \mathbf{x}_2 + \cdots + \mathbf{x}_k}{k},$$

and is the arithmetic mean position of all of the points.

We suggest a summary statistic ψ_j that is a normalised distance of the ObsDat (transformed under the PCA) $\mathbf{x}^{obs}$, from a cluster of points $S_j = \{\mathbf{x}_1^j, \mathbf{x}_2^j, \cdots, \mathbf{x}_{k_j}^j\}$, as follows.

We consider the 'distance' between $\mathbf{x}^{obs}$ and the centroid of the cluster, $\mathbf{c}^j$, divided by the root mean square within-cluster distance from the centroid, *in the direction of* $(\mathbf{x}^{obs} - \mathbf{c}^j)$.

To do so, we find the orthogonal projection matrix T_j such that $t_j(\mathbf{x}) = T_j\mathbf{x}$ is the map given by

$$t_j(\mathbf{x}) : \mathbb{R}^n \to \text{span}\{(\mathbf{x}^{obs} - \mathbf{c}^j)\}.$$

For each data point x_i^j, we consider the orthogonal projection $y_i^j = T_j\mathbf{x_i^j}, \quad i = 1, \cdots, k_j$, onto $(\mathbf{x}^{obs} - \mathbf{c}^j)$ (see Figure 5.16). We then calculate s_{y^j}, the root mean square distance of the projected points on $(\mathbf{x}^{obs} - \mathbf{c}^j)$, from the centroid.

The distance between the observed point $\mathbf{x}^{obs}$ and the centroid $\mathbf{c}^j$ is then normalised by s_{y^j}, which allows the spread of the points within the cluster, in the direction of $(\mathbf{x}^{obs} - \mathbf{c}^j)$, to be considered. We denote this normalised distance ψ_j', and we repeat this for each cluster $S_j, \quad j = 1, \cdots, m$. Finally, we divide each ψ_j' by $\sum_{\ell=1}^m \psi_j'$ so that $\psi_j \in [0, 1]$, $\forall j$. This process is defined in Algorithm 5.

We interpret ψ_j as 'the proportion of the total amount of orthogonally projected distance of ObsDat from the cluster centroids due to model M_j'. Clearly, the smaller the value of ψ_j, the closer the $\mathbf{x}^{obs}$ is to $\mathbf{c}^j$, with respect to the variability of points within S_j, and so we look for the smallest value of ψ_j. However, if no

one cluster seems most reasonable for $\mathbf{x}^{obs}$, then we should find that all values of ψ_j are roughly similar, and hence it could be that we never specified the correct model for $\mathbf{x}^{obs}$ in our candidate models.

Algorithm 5: Cluster Checking Algorithm.

Input: ObsDat = $\mathbf{x}^{obs}$, m clusters $S_1, S_2, \cdots, S_m$,
where $S_j = \{\mathbf{x}_1^j, \mathbf{x}_2^j, \cdots, \mathbf{x}_{k_j}^j\}$ is the j^{th} cluster.

for $j = 1, \cdots, m$ **do**
  Calculate $\mathbf{c}^j = \frac{\mathbf{x}_1^j + \mathbf{x}_2^j + \cdots + \mathbf{x}_{k_j}}{k_j}$;
  Let $A_j = (\mathbf{x}^{obs} - \mathbf{c}^j)$ and calculate the orthogonal projection matrix,
  $T_j = A_j \left(A_j^T A_j\right) A_j^T$.
  for $i = 1, \cdots, k_j$ **do**
    $y_i^j = ||T_j \mathbf{x}_i^j - \mathbf{c}^j||$
  end
  $s_{y^j} = \left[\frac{1}{k_j - 1} \sum_{\ell=1}^{k_j} \left(y_\ell^j\right)^2\right]^{\frac{1}{2}}$,
  $\psi_j' = \frac{||\mathbf{c}^j - \mathbf{x}^{obs}||}{s_{y^j}}$;
end
for $j = 1, \cdots, m$ **do**
  Calculate $\psi_j = \frac{\psi_j'}{\sum_{\ell=1}^m \psi_\ell'}$.
end

We performed Algorithm 5 for each ObsDat (Constant, Exponential and Migration Models) on the subset of our TrainDat and ABCDat with all Constant Model data removed (as in Figure 5.15). Unsurprisingly, we found evidence that the Exponential Model ObsDat belongs to the Exponential Model cluster, with $\psi_{Migration}$ roughly five times larger than $\psi_{Exponential}$ (see Table 5.6). Similarly we found evidence that the Migration Model ObsDat belongs to the Migration Model cluster, with $\psi_{Exponential}$ roughly seven times larger than $\psi_{Migration}$ (see Table 5.6).

For the Constant Model ObsDat we observe relatively similar values of $\psi_{Exponential} =$

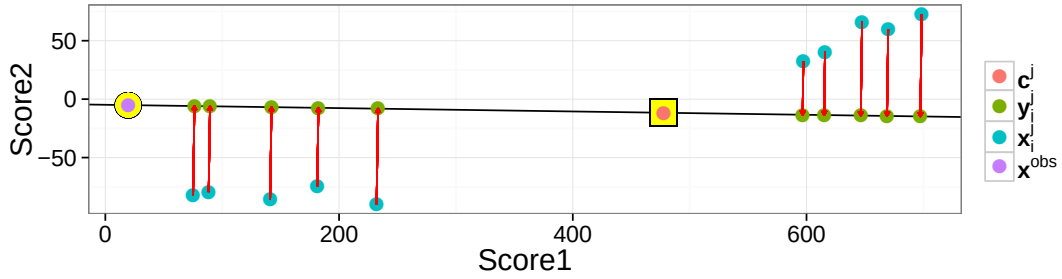


Figure 5.16: An example of Algorithm 5. The blue points $\mathbf{x}_i^j$ are orthogonally projected onto the line $(\mathbf{x}^{obs} - \mathbf{c}^j)$, and we call these $\mathbf{y}_i^j$ (the green points). We then calculate the root mean square of the distances of the $\mathbf{y}_i^j$ from the centroid $\mathbf{c}^j$ (the red point).

0.5188, and $\psi_{Migration} = 0.4812$ (see Table 5.6). This suggests that the Constant Model Obsdat falls relatively evenly between the two clusters, and that it might be worth investigating whether or not we have specified a sensible model for the Constant Model Obsdat (see Figure 5.17).

ObsDat	$\psi_{Exponential}$	$\psi_{Migration}$
Constant	0.5188	0.4812
Exponential	0.1331	0.8669
Migration	0.8910	0.1090

Table 5.6: The observed values of ψ for each ObsDat.

We suggest visualising the values for $P(M_j\,|\mathbf{X})$ obtained from the MLR classification with the associated values of $1 - \psi_j$ in the following way. We can think of $1 - \psi_j$ as 'the proportion of the total amount of orthogonally projected distance of ObsDat from the cluster centroids that is the not due to model M_j'. If model M_j is the correct model, we would expect that the proportion of the orthogonally projected distance of ObsDat from the cluster centroids (ψ_j) should be small compared to ψ_i, $i \neq j$, (as $\mathbf{c}^j$ should be relatively 'close' to $\mathbf{x}^{obs}$). Hence

$(1-\psi_j)$ should be 'close' to one. Similarly, if model M_j is the correct model, then $P(M_j\,|\mathbf{X})$ should also be 'close' to one.

Since $\psi_j \in [0,1]$, then $(1-\psi_j) \in (0,1)$ for all j (and clearly $P(M_j\,|\mathbf{X}) \in (0,1)$ for all j), we plot the MLR probabilities for each model against the values of $(1-\psi_j)$, and look for points in the top, right corner (approaching (1,1)). We call these the 'two-way model fit' plots, and examples are given in Figure 5.18.

Similarly, note that in Figure 5.18 (a), when we have model misspecification for the ObsDat, neither of the simulation model data points has moved toward the top corner. However, in Figures 5.18 (b) and (c), when the relevant model is specified for the ObsDat, the correct simulation model data point has moved toward the top corner.

Finally, we look at a specific case where data has been misclassified by our MLR classification scheme. In Figure 5.19 we analyse a specific simulation from the Constant Model ABCDat. Our MLR soft classifications are 'Exponential' with probability 0.599, and 'Constant' with probability 0.4. That is, our MLR classifier would hard classify this data point as the product of an Exponential Model, and looking at Figure 5.19, it appears to be on the edge of both clusters.

While it seems that the data is more likely to be Exponential, when we look at our values for ψ (see Table 5.7), we see that $\psi_{Constant}$ is almost half the value of $\psi_{Exponential}$. That is, it appears that classifying the data as Constant fits better for the PCA cluster checking algorithm. By considering the values of ψ, we identify a spurious hard classification, that was claiming that the incorrect model was 'more likely' than the correct model.

We point out that by plotting $1-\psi$ visually, we should be careful when considering the two-model model fit plot for a large number of models. Consider,

$$\sum_{i=1}^{n} \psi_i = 1 \implies \sum_{i=1}^{n} 1-\psi_i = n-1.$$

We hope some model j is 'best', and this implies that $1-\psi_j \approx 1$. Hence, it must

ObsDat	$\psi_{Constant}$	$\psi_{Exponential}$	$\psi_{Migration}$
Constant	0.1445	0.2797	0.5757

Table 5.7: The observed values of ψ for the misclassified Constant data from Figure 5.19.

be that the remaining $n-1$ values of $1-\psi_i$ sum to approximately $n-2$. If all of the remaining models were equally bad, then the next largest value of $1-\psi_i$ will be

$$\frac{\sum_{i\neq j} 1-\psi_i}{n-1} = \frac{n-2}{n-1},$$

and this would be the minimum value that $1-\psi_i$ could take. Hence, the maximum difference between $1-\psi_j$ and $1-\psi_i$ is

$$\begin{aligned}(1-\psi_j)-(1-\psi_i) &\approx 1-\frac{n-2}{n-1}\\ &= \frac{1}{n-1}\\ &\to 0,\end{aligned}$$

as $n\to\infty$. That is, the maximum vertical distance between the correct model, and the next 'best' model becomes smaller the more candidate models we nominate, and we must take this into account when considering these values. For this reason, we draw horizontal and vertical lines that are $\frac{1}{n-1}$ apart for perspective when analysing these two-way model fit plots.

5.5 A Data Driven Algorithm for Model Selection and Parameter Estimation

In Chapter 6, we shall make inferences about a new ObsDat. We obtain the data from a new family of simulation techniques called 'forward simulation'. We discuss

both the method by which the new Obsdat is generated, and the motivation for doing so, and we follow the process outlined below.

Data Driven Model Selection and Parameter Estimation Algorithm

1. From a candidate model set $\mathbf{M} = \{M_1, \ldots, M_q\}$, use MLR classification (see Section 5.4), PCA visualisation and The Cluster Checking Algorithm 5 to select a most suitable model for parameter estimation (if one exists in the candidate model set).

2. Use Algorithm 4 to perform parameter estimation via ABC under the model selected in step 1.

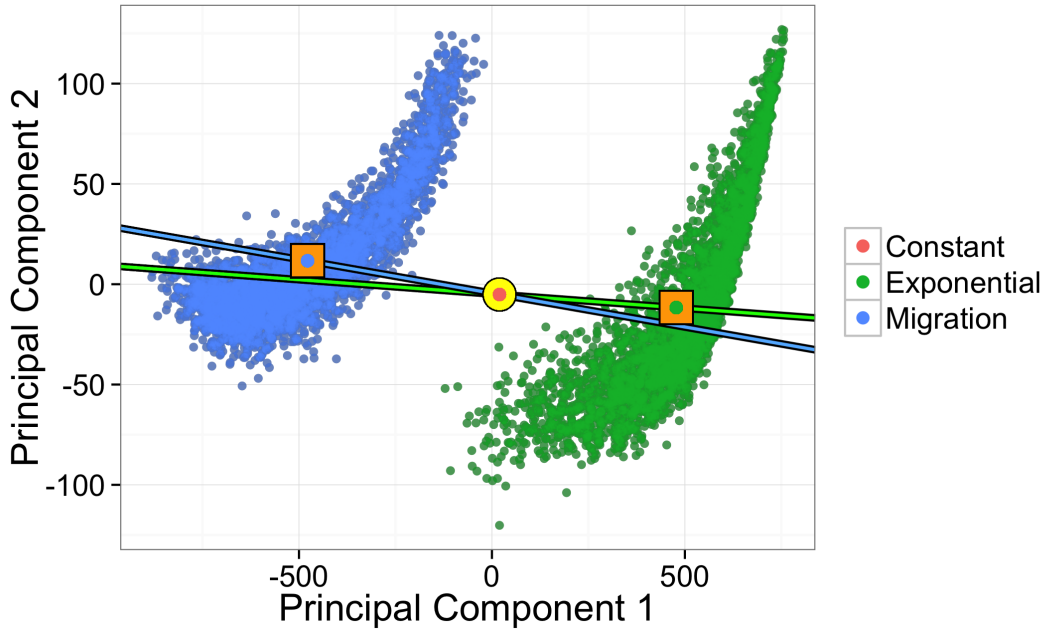


Figure 5.17: The centroids (square points) for the Exponential Model cluster (green) and the Migration Model cluster (blue) relative to the Constant Model ObsDat (red circle). The vectors on to which we orthogonally project are given for each cluster.

Figure 5.18: Two-way model fit plots for the combined Exponential and Migration TrainDat with (a) the Constant ObsDat, (b) the Exponential ObsDat, and (c) the Migration ObsDat.

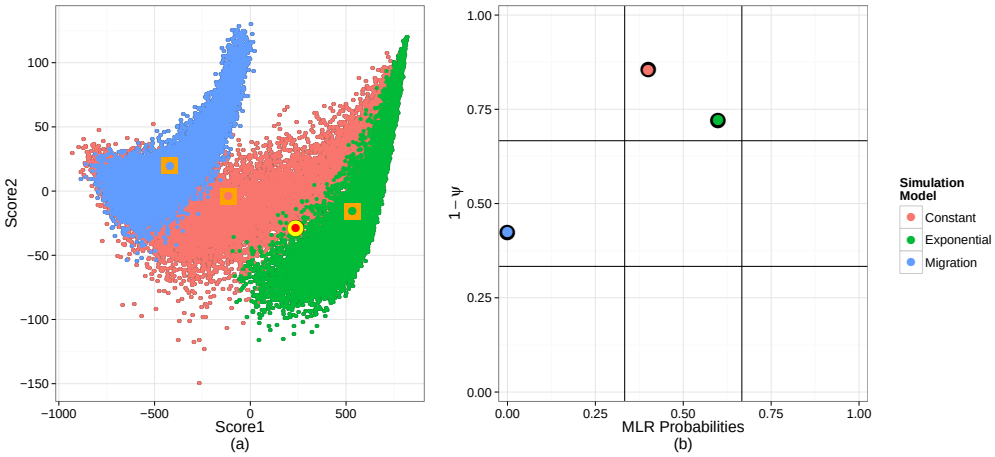


Figure 5.19: (a) A PCA plot of the first two principal components for the combined Constant, Exponential and Migration TrainDat with centroid clusters (square data points) and a misclassified ObsDat (the circle data point) and (b) a two-way model fit plot for a Constant Model Simulation.

Chapter 6

Bottleneck Data Analysis

The initial motivation for our research was the question of whether one could better identify the time of a bottleneck event, and the effective population sizes before and after such an event, than current methods. Recall from Chapter 2 that a bottleneck event is one in which a breeding population undergoes a sudden reduction in effective breeding population size from (the 'base size') $N(t_0)$ to (the 'bottleneck size') $N(0) = \beta N(t_0)$ at event time t_0 (where β is the proportion of the population remaining after the crash).

For this final analysis, we investigate how the algorithm we have suggested for model selection and parameter estimation performs on this model of population dynamics. We again simulate the data, and we do this for several reasons.

First, the method we have developed is in the very early stages of development. As such, we have not yet developed methods for dealing with missing data (data where positions in the sequence are unknown), and hence we are restricted to 'perfect data' for the moment.

Second, we acknowledge the computational limitations of this project, and hence we are restricted to exploring parameter spaces of relatively low dimension. While this is not a short coming of our method, it is a necessary condition for any data

analysis we wish to perform.

Third, it is sensible to assume that sequences from before, after, and possibly during events must be obtained if analyses are to have any chance of identifying these events. Only sampling modern samples can not be relied upon to recover past population events [19]. If we are to investigate a data set for which an event has occurred, such as a bottleneck event as was initially discussed, we require a data set with samples at different times through the past.

That is, we require data with the following properties:

1. No missing sequence characters.
2. A parameter space of low dimension.
3. Multiple sampling times.

Since we have been unable to find a data set that met these three criteria, we simulated a new ObsDat. Importantly, we simulate the Obsdat from a different family of simulation methods called 'forward simulation', rather than the coalescent simulations we have been using (recall, due to computational efficiency, coalescent simulation is used for TrainDat and ABCdat). A key difference between the simulation methods is that coalescent simulation looks 'backward in time' to build a genealogy, before applying mutations to the genealogy. Forward simulation only looks 'forward in time', and creates a genealogy with mutations, sequentially, generation by generation.

6.1 Forward Simulation

We begin with the initial population at ξ generations before present (bp), and this generation has a pre-specified distribution of sequences (often this distribution is the equilibrium distribution for the model of sequence evolution, or all individuals have identical sequences). We initially consider the process of inheritance

(the choice of 'children'), and then consider the independent process of mutation separately. Keep in mind that we are always looking forward in time during this process, and so each successive generation is closer to the present than the previous generation.

In Section 2.1, we describe the Wright-Fisher model of reproduction as non-overlapping (equally sized) generations undergoing random mating. We relax the assumption of equal size for each generation, and define the population size i generations in the past as $N(i)$. Now each child in generation $(i-1)$ bp has an equal probability of $1/N(i-1)$ of being selected to inherit the sequence from any one of the parents in the generation i bp. For each individual in the generation i bp, we randomly select a child from the next generation, and assign a copy of the sequence to the child.

We need only now consider the independent process of mutation, given the (per site per individual per generation) mutation rate μ, and the sequence length ℓ. The number of nucleotide mutations, η_i, for the population i generations in the past can be modelled using a Poisson distribution with parameter $\mu \times \ell \times N(i)$. Hence, once the process of inheritance is complete, we sample $\eta_i \sim \text{Po}(\mu \times \ell \times N(i))$, and then randomly select (without replacement) η_i sites from the possible $\ell \times N(i)$ sites in the population. We then alter these sites according to the mutation process. For a visual summary of this process, see Figure 6.1.

We use the software TreeSimJ for the forward simulation of the Bottleneck ObsDat with the following parameters [31]:

1. A mutation rate of $\mu = 10^{-6}$ per site per individual per generation.

2. A sequence length of $\ell = 1000$bp.

3. The Jukes-Cantor model of mutation.

4. Neutral fitness site model (no effect from natural selection).

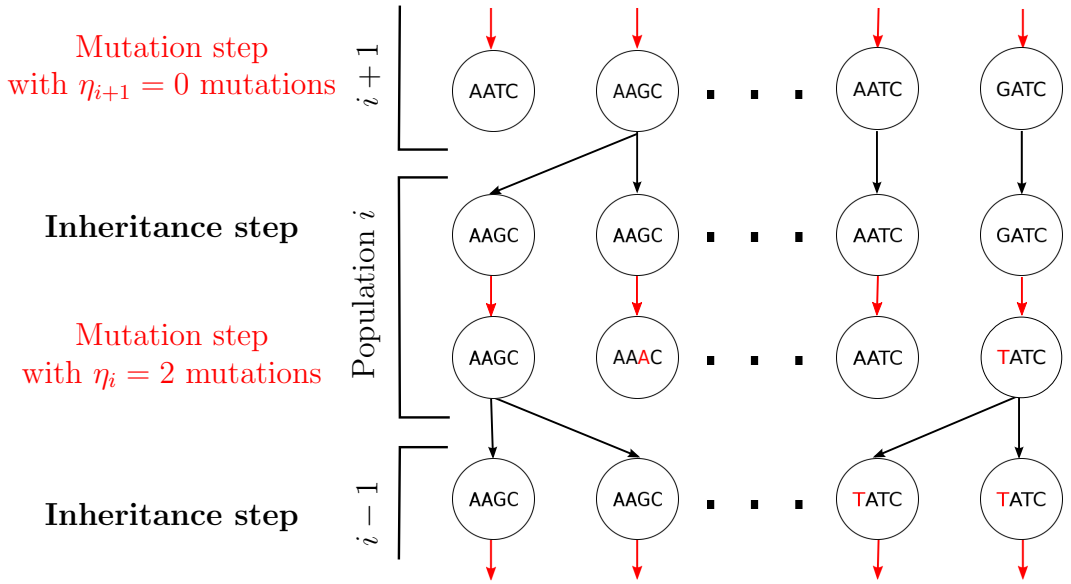


Figure 6.1: The forward simulation process for the population i generations in the past, with sequence length $\ell = 4$. The black arrows indicate an inheritance step, and the red arrows indicate a mutation step.

5. Base Size $N(t_0) = 15,000$ breeding individuals.

6. Proportional population decrease $\beta = 0.1$ (and hence bottleneck size $N(0) = 1,500$ breeding individuals).

7. Bottleneck frequency 9,500 generations.

8. Bottleneck duration 1,500 generations.

9. 75 sequences sampled at time 0, 50 sequences sampled at times 1000 and 2000 generations in the past.

10. Burn-in 10,000 generations.

11. $\xi = 100,000$ generations.

The 'burn-in', 'bottleneck frequency', and the 'bottleneck duration' input parameters are employed by TreeSimJ in the following way. For burn-in, the inheritance-mutation steps are repeated 10,000 times for a constant population of size $N(0)$ (and no data is recorded for efficiency). The population then increases to size $N(t_0)$ (after 1500 generations), and remains at size $N(t_0)$ for $9500 - 1500 = 8000$ generations. The population size crashes to size $N(0)$ for 1500 generations, before returning to size $N(t_0)$, and repeating this step until ξ generations have passed. The population data we actually recover from TreeSimJ is shown in Figure 6.2, with DNA sampled every 1000 generations, and we only consider the blue shaded area. This sampling scheme resulted in a bottleneck event time of 600 generations bp. The extracted population dynamics are presented in Figure 6.3.

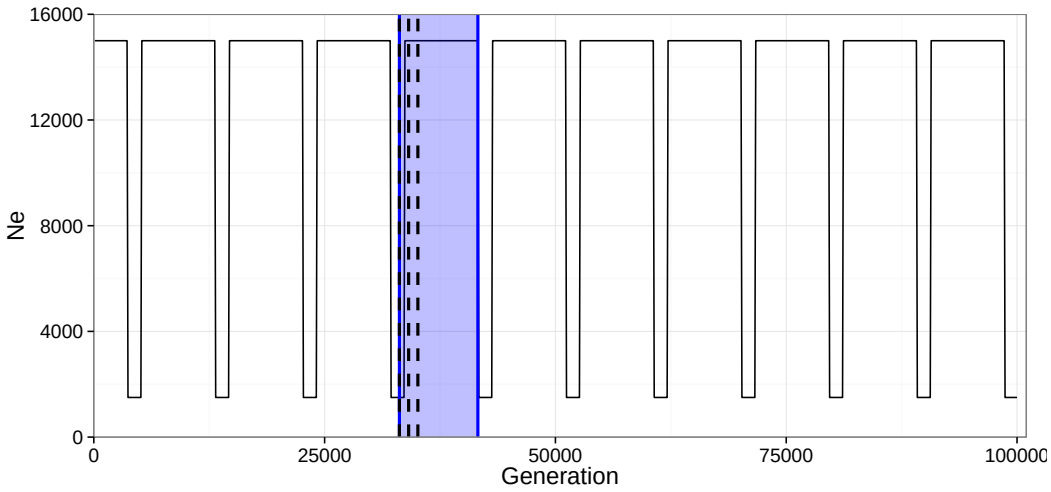


Figure 6.2: The full TreeSimJ data range, with the subset of generations (shaded blue) we keep for the Bottleneck ObsDat.

The TrainDat and ABCDat for the Bottleneck Model have the parameters, and prior distributions listed in Table 6.1. The Bottleneck Model is parameterised differently in BayeSSC, than it is in TreeSimJ. For simulating in BayeSSC we specify a distribution for the bottleneck size $N(0) = \beta N(t_0)$, and a proportional increase (β^{-1}) in size to the base size $N(t_0)$. To avoid confusion, we will report on the posterior distributions for $N(0)$, $N(t_0)$ and t_0 only.

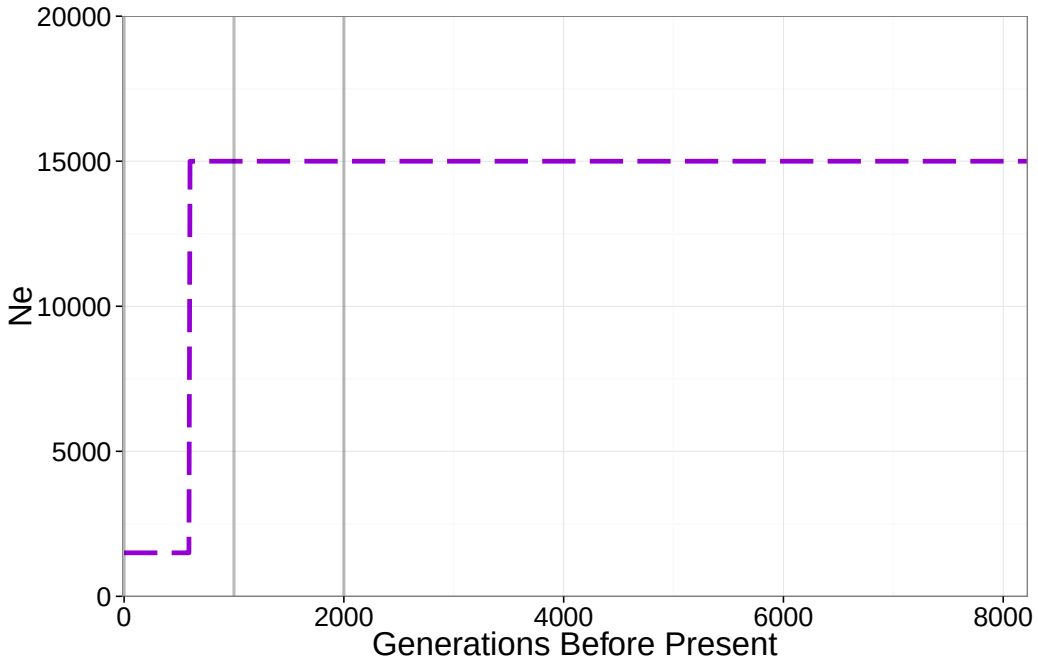


Figure 6.3: The extracted TreeSimJ data range kept for the analysis. The solid grey lines represent sampling times, and the purple dashed line is the true effective population size.

Note that we have two ABCDats, ABCDat_1 and ABCDat_2. We do this to better explore the posterior distribution for t_0, and this will be explained later in this chapter.

6.2 MLR Classification Step

For the candidate model set we consider the Bottleneck, Constant, Exponential and Migration Models as presented in Tables 6.1 and 4.2 for the TrainDat (and the MLR classification). See Appendix A.8.

The ObsDat is predicted to be the result of a Bottleneck Model of population dynamics with probability 0.9999 (see Table 6.2), and hence the MLR model would hard classify the ObsDat as Bottleneck Model data.

Data Set	Parameter values and Prior Distributions
ObsDat	$N(0) = 1500$ $t_0 \sim U(1, 4000)$ $\beta^{-1} = 10$
TrainDat	$N(0) \sim U(250, 15000)$ $t_0 \sim U(1, 4000)$ $\beta^{-1} \sim U(1, 20)$
ABCDat_1	$N(0) \sim U(500, 7500)$ $t_0 \sim U(50, 2000)$ $\beta^{-1} \sim U(5, 15)$
ABCDat_2	$N(0) \sim U(1203, 2414)$ $t_0 \sim U(50, 2000)$ $\beta^{-1} \sim U(3.41543, 13.99880)$

Table 6.1: Parameter values and prior distributions for the Bottleneck Models.Common to all simulations: Sequence length $\ell = 1000$ bp, Mutation Model: Jukes-Cantor and a mutation rate $\mu = 10^{-6}$ per site per individual per generation.

	$P(M_C \mid \mathbf{X})$	$P(M_E \mid \mathbf{X})$	$P(M_M \mid \mathbf{X})$	$P(M_B \mid \mathbf{X})$
$\mathbf{X} = \mathbf{X}_B$	3.8251×10^{-05}	6.2625×10^{-6}	1.0987×10^{-30}	0.9999

Table 6.2: Predicted model classification probabilities for the the ObsDat $\mathbf{X}_B$ where M_C,M_E,M_M and M_B are the Constant, Exponential, Migration and Bottleneck Models respectively.

We perform a Principal Component Analysis of the combined TrainDat (used to train the MLR classification model), and a scatterplot of the first two principal

components is presented in Figure 6.4, with the Obsdat in the yellow circle.

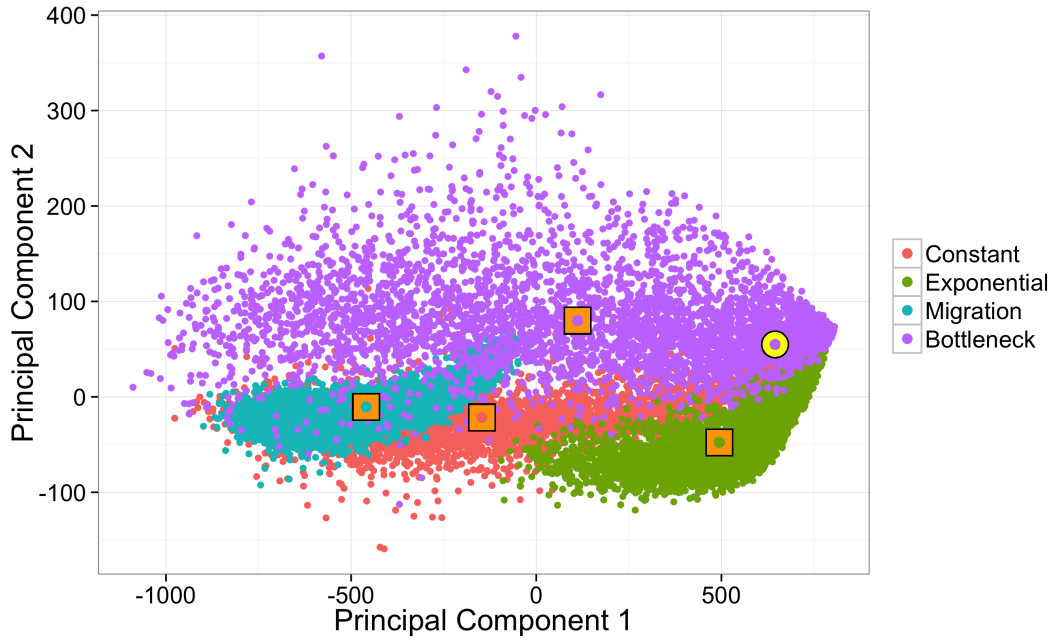


Figure 6.4: The first two principal components for the combined TrainDat data set, coloured by the model under which simulation was performed. The yellow circular data point is the ObsDat, and the orange square data points are the cluster centroids.

The ObsDat has clustered correctly with the Bottleneck Model data cluster visually, and this is consistent with the data correctly classifying as Bottleneck Model data via the MLR method. However, there is some crossover between the Bottleneck, Constant and Exponential data near the Obsdat. This crossover is due to the fact that an Exponential Model with growth rate $K = 0$, and a Bottleneck Model with $\beta = 1$ are effectively a Constant Model, and these values for K and β were included in the prior distributions. Similarly, a Bottleneck Model with $t_0 \approx 1$ will appear approximately Constant. The closeness of these three clusters to the ObsDat is highlighted in the values obtained from the cluster checking algorithm (Algorithm 5).

We obtained observed values for ψ as presented in Table 6.3, and immediately we

ObsDat	$\psi_{Constant}$	$\psi_{Exponential}$	$\psi_{Migration}$	$\psi_{Bottleneck}$
Bottleneck	0.1702	0.2122	0.5114	0.1062

Table 6.3: The observed values of ψ for ObsDat.

observe that the Migration Model centroid accounts for more than half (51.14%) of the total normalised orthogonally projected distance from the ObsDat. Similarly the Exponential and Constant Models account for 17.02% and 21.22% of the total normalised orthogonally projected distance, where the Bottleneck Model accounts for only 10.62% of the total normalised orthogonally projected distance. This evidence for the Bottleneck Model is further supported when we consider the associated MLR soft classification probabilities. From Figure 6.5 we observe the Bottleneck Model point tending toward the top, right corner. While the Constant and Exponential Models both have relatively large values for $1 - \psi$, the Bottleneck Model has the the largest value for $1 - \psi$, and a significantly larger MLR probability.

Given the results of the MLR classification, inspection of the PCA visualisation, and the results of the cluster checking algorithm and two-way model fit plot, we select the Bottleneck Model for the purpose of parameter estimation.

6.3 Parameter Estimation Step

Step 0: Obtaining ObsDat.

Recall we have a population of constant size 15,000 breeding individuals that abruptly drops to a constant size of 1,500 breeding individuals 600 generations before present. We sample 75, 50 and 50 sequences at 0, 1000, and 2000 generations in the past respectively (see Table 6.1).

Step 1: Producing TrainDat and defining the Linear Model

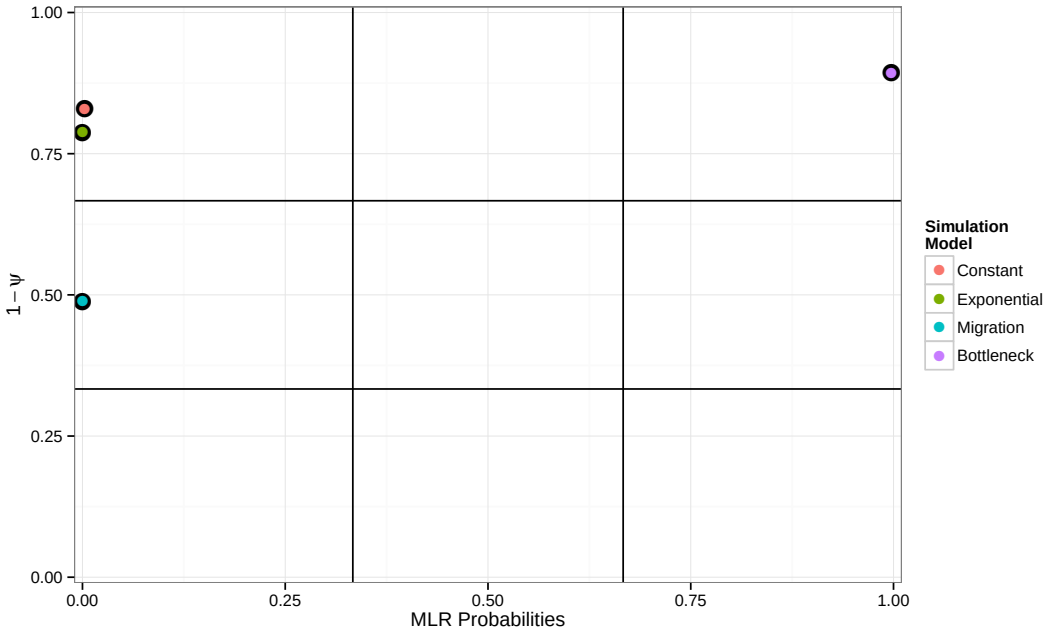


Figure 6.5: Two-way model fit plot for the combined Bottleneck, Constant, Exponential and Migration TrainDat with the Bottleneck ObsDat.

For the TrainDat for the Bottleneck Model we simulated $150,000$ realisations of the Bottleneck Model with a prior distribution of $N(0) \sim U(250, 15000)$ and $t_0 \sim U(1, 4000)$. For the proportional increase in population size, we have prior distribution $\beta^{-1} \sim U(1, 20)$, and this corresponds to $N(t_0) \in (250, 3 \times 10^5)$ (although this does not correspond to a uniform distribution for $N(t_0)$). Using TrainDat, we fitted a linear regression model to the training set on the transformed bottleneck size.

The fitted linear model for the bottleneck size was of the form

$$\hat{N(0)}^{\lambda^1_{M_4}} = \beta_0 + \sum_{i=1}^{30} \beta_i s_i,$$

where the observed Box-Cox transformation parameter (see Figure 6.6) for the bottleneck size was $\lambda^1_{M_4} = 0.3395$ (see Section A.6 for the list of the 30 retained summary statistics and associated coefficients).

Almost every candidate summary statistic was retained, with the exception of H_1,

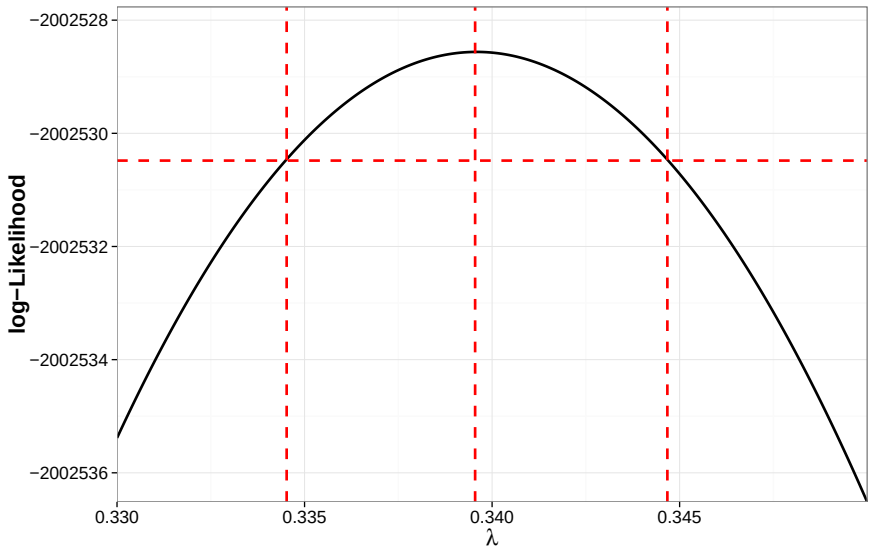


Figure 6.6: Box-Cox transformation profile likelihood with 95% confidence interval for $\lambda^1_{M_4}$.

H_2, h_1n h_2 and $p_{(2,1)}$. It appears as if the difference in haplotype distribution between the samples at 1000 and 2000 generations in the past is of little importance, and this makes sense since the population is of a constant size between these points (so haplotype distributions will not differ much). Information about the haplotype distribution at these sampling times is recovered from $H_{\{0,1,2\}}$ and $h_{\{0,1,2\}}$, and from comparisons of stat group 0 and stat groups 1 and 2 through H_T and $\bar{H}_S$ (see Figure 6.9).

The fitted linear model for the event time was of the form

$$\hat{t_0}^{\lambda^2_{M_4}} = \beta_0 + \sum_{i=1}^{28} \beta_i s_i,$$

where the observed Box-Cox transformation parameter (see Figure 6.7) for the bottleneck size was $\lambda^2_{M_4} = 0.7796$ (see Section A.6 for the list of the 28 retained summary statistics and associated coefficients).

The summary statistics retained for the linear model for the event time t_0 are similar to those retained for the linear model for the event time t_0. The exceptions are that H_1 is now retained, but $\bar{H}_S^{(0,1)}$, $\pi_{\{0,1,2\}}$ and $H_T^{(0,2)}$ are not. An argument,

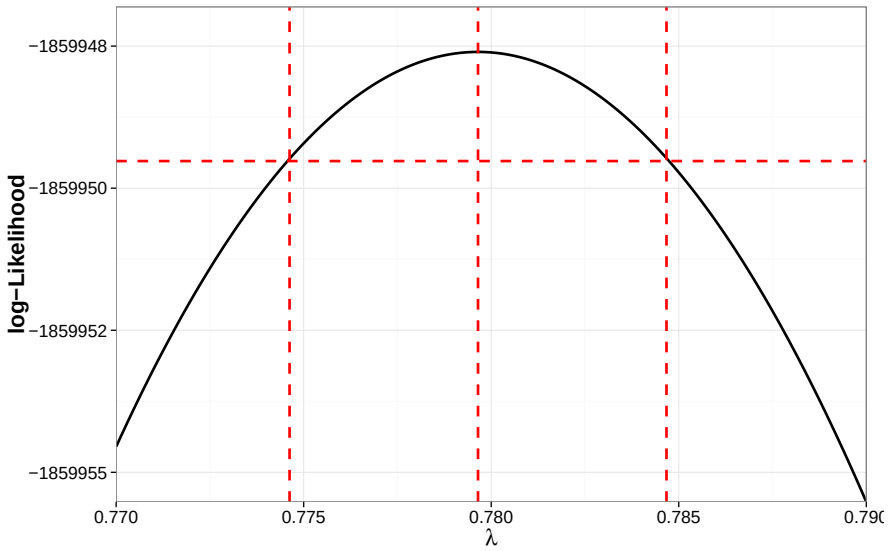


Figure 6.7: Box-Cox transformation profile likelihood with 95% confidence interval for $\lambda^2_{M_4}$.

similar to the argument presented for the pattern of retained summary statistics for the linear model of $N(0)$, exists here, since again the difference in haplotype distributions for stat groups 1 and 2 should not differ (see Figure 6.9).

The fitted linear model for the base size was of the form

$$\hat{N(t_0)}^{\lambda^3_{M_4}} = \beta_0 + \sum_{i=1}^{30} \beta_i s_i$$

where the observed Box-Cox transformation parameter (see Figure 6.8) for the bottleneck size was $\lambda^3_{M_4} = 0.6313$ (see Section A.6 for the list of the 30 retained summary statistics and associated coefficients).

For the linear model for the base size $N(t_0)$, the retained summary statistics do not appear to display any meaningful pattern. As is the case in all analyses we present, Tajima's D statistic is always retained, and this is probably because it is the only summary statistic that represents selection neutrality, instead of genetic diversity. Additionally, nucleotide diversity (π) is retained for all stat groups.

A comparison of the groups at 1000 and 2000 generations in the past would seem

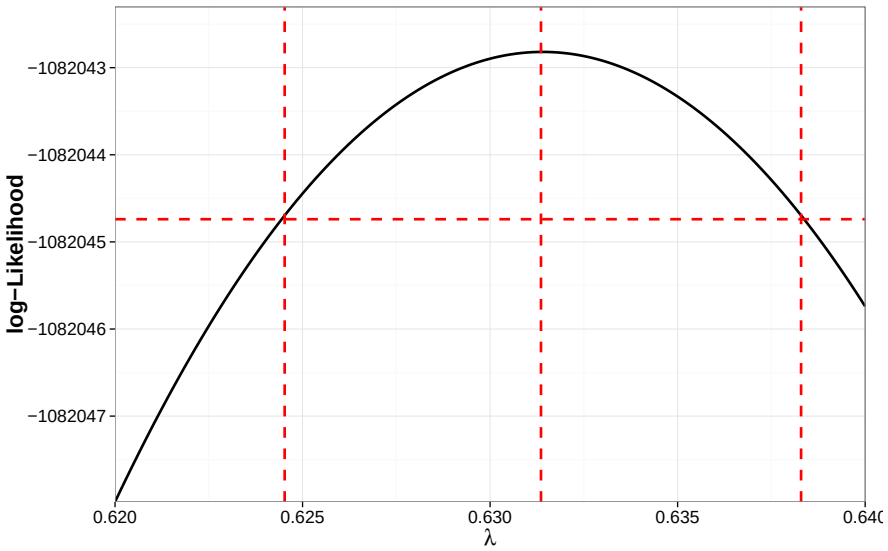


Figure 6.8: Box-Cox transformation profile likelihood with 95% confidence interval for $\lambda^3_{M_4}$.

most beneficial when detecting the population size around these times. Some of the retained summary statistics retained reflect this, as $H_T^{(1,2)}$ and $F_{ST}^{(1,2)}$ are retained, when $H_T^{(0,2)}$, $F_{ST}^{(0,1)}$ and $F_{ST}^{(0,2)}$ are not. The only departure from this theme is that $p_{(1,2)}$ (the number of alleles unique to 1000 generations in the past when compared to 2000 generations in the past) is not retained, although we note that $p_{(2,1)}$ is retained (see Figure 6.9).

We argue that the ABC analysis should be done in a two-step process for this ObsDat. Consider that under the Wright-Fisher Model assumptions (which are satisfied under this simulation model), the true population size (N_e) is a function of the true genetic diversity (θ) in the population, and is given by

$$\theta = 2N_e\mu,$$

for a haploid population (where μ is the mutation rate per site per individual per generation).

It makes sense that we estimate the population sizes ($N(0)$ and $N(t_0)$) from the genetic diversity measures (the summary statistics) first, before we attempt to find

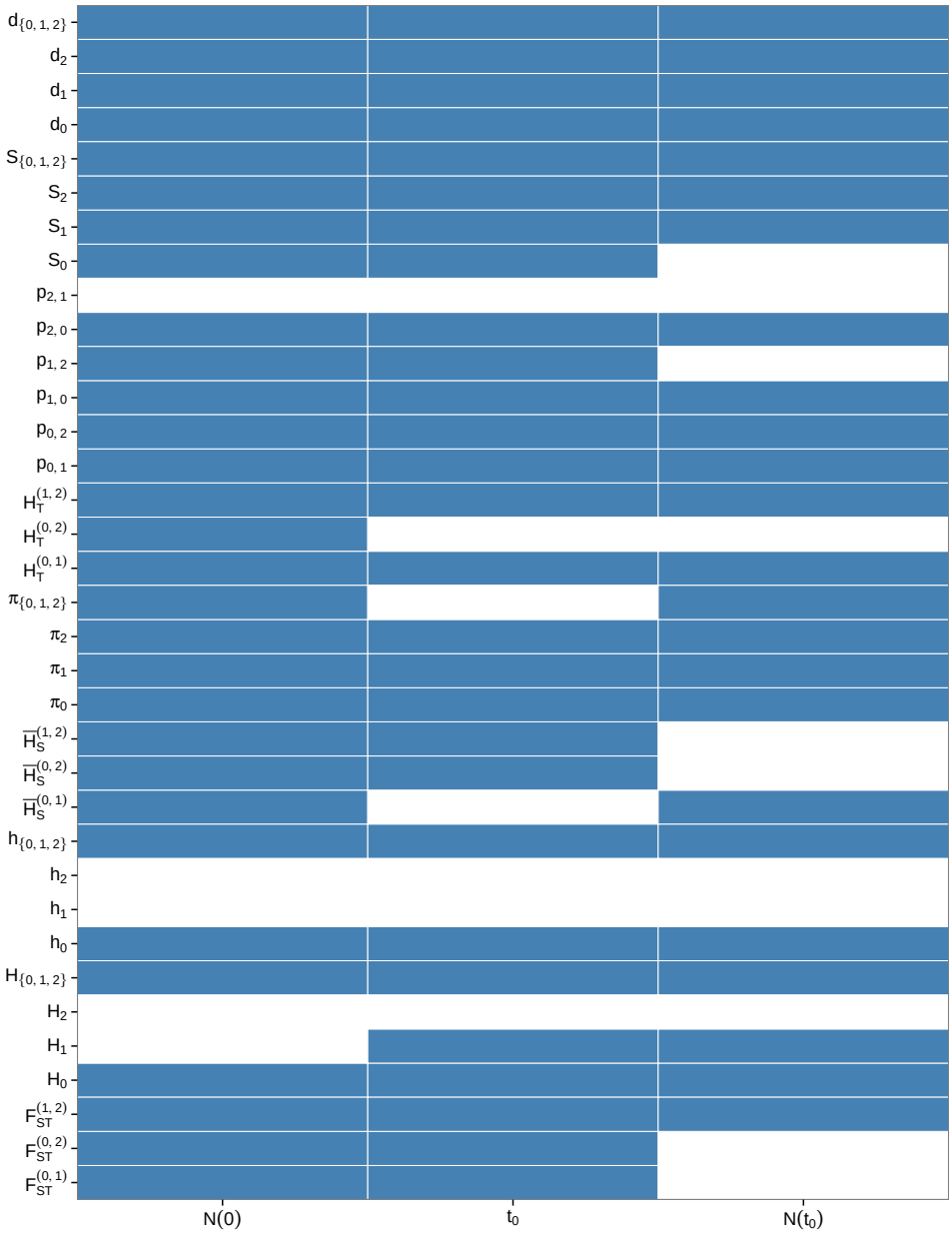


Figure 6.9: Retained summary statistics for each transformed parameter linear model for the Bottleneck Model Analysis.

at what time the population size crash from $N(t_0)$ to $N(0)$ occurs. Hence we begin by performing the ABC step (Step 2 from Section 4.2) on ABCDat_1 to determine a sample from the posterior distributions of $N(0)$ and $N(t_0)$. From these samples of the posterior distributions, we obtain median values ($M_{N(0)}$ and $M_{N(t_0)}$) and sample standard deviations ($s_{N(0)}$ and $s_{N(t_0)}$).

We create a second data set, ABCDat_2, where we have the restricted posterior distributions,

$$N(0) \sim U(M_{N(0)} - s_{N(0)}, M_{N(0)} + s_{N(0)})$$

and

$$N(t_0) \sim U(M_{N(t_0)} - s_{N(t_0)}, M_{N(t_0)} + s_{N(t_0)}),$$

and we repeat the ABC step, this time on the restricted ABCDat_2, and make inferences on the posterior distributions recovered from this step of the analysis.

Step 2(a): ABC comparing Obsdat to ABCDat_1

For the ABCDat_1 we simulated $150,000$ realisations of the Bottleneck Model with a prior distribution for $N(0) \sim U(500, 7500)$, $t_0 \sim U(50, 2000)$ and $\beta^{-1} \sim U(5, 15)$ and retained the 500 'closest' simulations for the untransformed posterior estimates.

This led to a posterior median bottleneck size of $M_{N(0)} = 1808$, with associated sample standard deviation $s_{N(0)} = 605.11$, and a posterior median base size of $M_{N(t_0)} = 12790$, with associated sample standard deviation $s_{N(0)} = 4568.64$.

Step 2(b): ABC comparing Obsdat to ABCDat_2

For the ABCDat_2 we simulated $350,000$ realisations of the Bottleneck Model with a prior distribution for $N(0) \sim U(1203, 2414)$, $t_0 \sim U(50, 2000)$ and $\beta^{-1} \sim U(3.41, 13.99)$ and retained the 500 'closest' simulations for the untransformed posterior estimates.

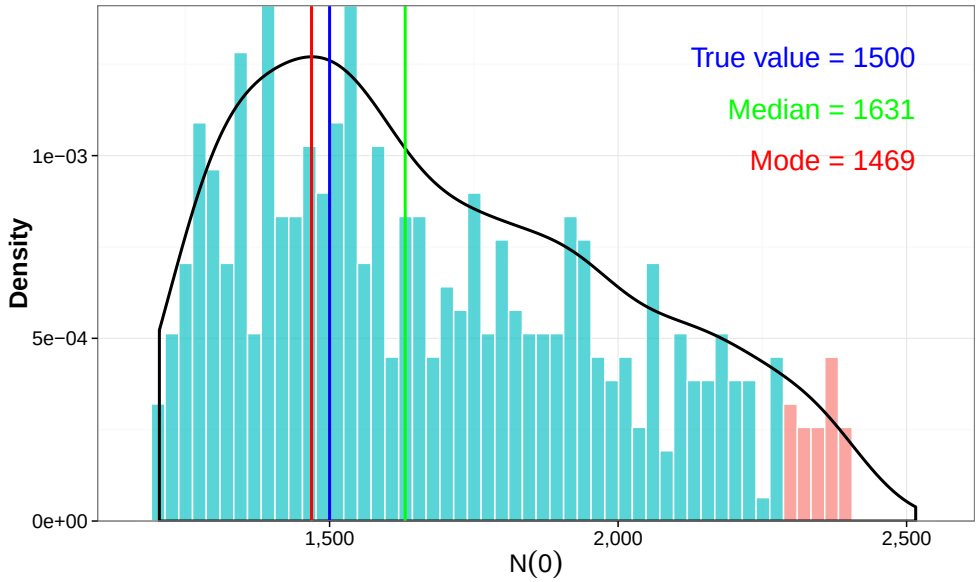
We first discuss the posterior distributions for $N(0)$ and $N(t_0)$ and compare these

with the results of the corresponding BEAST analysis from 0 generations bp and at the T_{MRCA} estimate from BEAST, 9747 generations bp (either end of the plotted BEAST BSP). We then compare our results for T_{MRCA} with the results from the BEAST analysis. Finally, we report our findings for t_0, and investigate possible values of t_0 from the BEAST analysis, and compare our results.

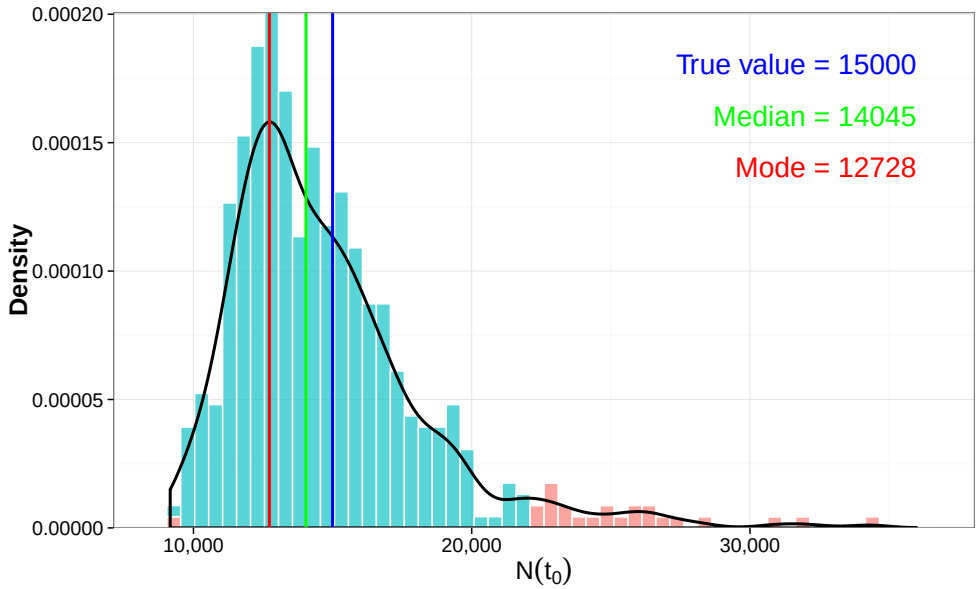
We estimate a median bottleneck size of $N(0) = 1631$, with a 95% probability interval $(1205, 2283)$ breeding individuals (see Figure 6.10). The width of the 95% probability interval for the estimated effective population size is then 1078 breeding individuals. The BEAST analysis returns an estimate of 1337.607, with a 95% probability interval $(139, 9534)$ breeding individuals, with interval width 9395. Our median posterior estimate for $N(0)$ is closer to the true value of 1500 than the estimate from BEAST, and the 95% probability interval from BEAST is 8.71 times larger than the 95% probability interval from our analysis.

Similarly, we estimate a median base size of $N(t_0) = 14045$, with a 95% probability interval $(9272, 21794)$ breeding individuals (see Figure 6.10). The width of the 95% probability interval for the estimated effective population size is then 12522 breeding individuals. The BEAST analysis returns an estimate of 10239.33, with a 95% probability interval $(1686, 19099)$ breeding individuals, with interval width 17413. For this parameter, our median posterior estimate for $N(t_0)$ is again closer than the BEAST results to the true value of 15,000. We note that the 95% probability interval from BEAST is only 1.4 times larger than the 95% probability interval from our analysis.

We estimate a T_{MRCA} of 22637 generations bp, with a 95% probability interval $(7902, 52848)$ generations bp (see Figure 6.13). The BEAST analysis reports a median T_{MRCA} of 13143 generations bp, and a corresponding 95% probability interval of $(9747, 16966)$ generations bp. Both probability intervals contain the true T_{MRCA} of 16313, and BEAST does better than our analysis with respect to interval width. However, recall how we collect data from the raw TreeSimJ output (see



(a)



(b)

Figure 6.10: Posterior samples for (a) the bottleneck size $N(0)$ and (b) the base size $N(t_0)$ for the Bottleneck Population Model with the true value, the posterior mode of the kernel density estimate and the posterior median indicated. The 95% probability interval is highlighted light blue. The kernel density estimate is given in black.

Figure 6.2). We are able to record the sample T_{MRCA} for each generation in the simulation, and this is the value presented here. Note that the $T_{\text{MRCA}} > 9500$, which was the 'repeat time' for bottleneck events in the simulations. Hence, the true simulation data has undergone another bottleneck event at 8600 to 10100 generations bp (and recall coalescence occurs more rapidly as $N(t) \to 0$).

Since we did not include this past event in our coalescent simulation specifications, we expect the coalescent simulations will over-estimate the T_{MRCA}. However, given the forward simulation is of size $N(t_0) = 15000$ for 8000 generations before the bottleneck event, the population allelic frequencies again come to an equilibrium, and so this past event will have little to no effect on any of the observed sample statistics (except the T_{MRCA}).

From Figure 6.11, the BEAST analysis appears to suggest two population declines; a sudden decline at about 1000 generations bp, and then again slowly until 0 generations bp. Clearly the population size dynamics recovered from BEAST are driven by sampling events, and this makes a comparison of parameter estimates for t_0 difficult.

Our analysis reports a posterior distribution for t_0 with median 581 generations bp, which is within 0.051 (posterior) sample standard deviations of the true value of $t_0 = 600$ generations bp. However, we report a 95% probability interval of $(58, 1299)$. This is relatively wide given that we expect our measures of genetic diversity (and hence population size) at sampling times would force the bottleneck event to occur within 0 and 1000 generations before present. However, when we look at the posterior distribution, and restrict $t_0 \in (1000, 1299)$ (t_0 in the 95% probability interval *and* greater than 1000), we observe two things.

First, the restricted subset of the posterior distribution of t_0 is strongly positively skewed (towards 1000), and in fact has a sample median of 1050 generations bp. Second, the distribution of the bottleneck sizes ($N(0)$) associated with the restricted values of t_0 have a sample median of 2033, and a sample standard deviation

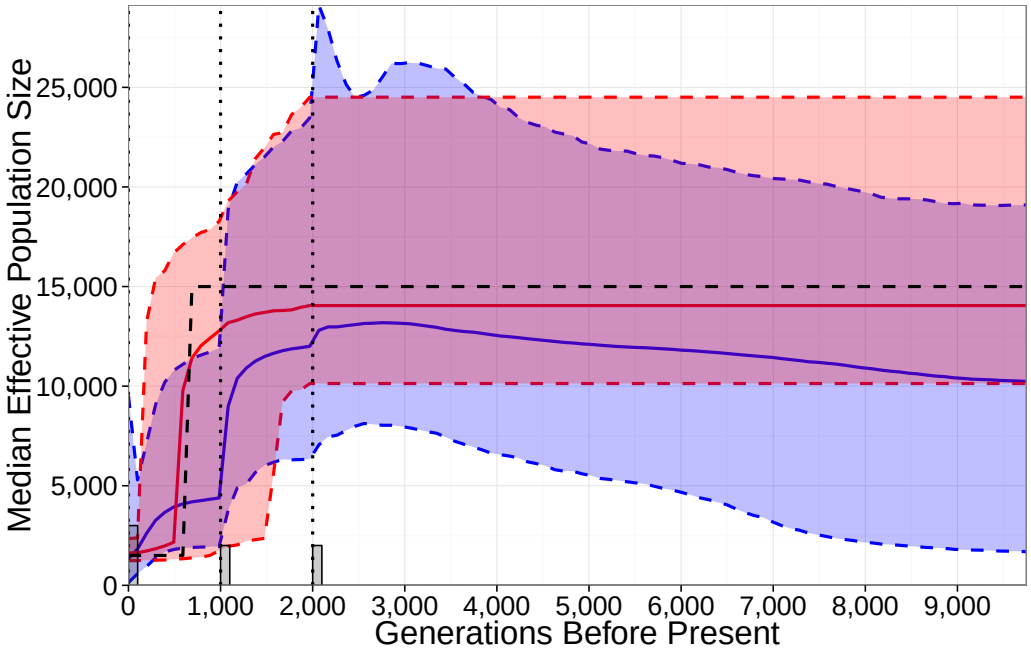


Figure 6.11: A comparison of our median effective population size estimates (in red) with those obtained from BEAST (in blue). The black dashed line is the true effective population size. The red and blue shaded areas are the 95% probability regions.

of 240 breeding individuals. That is, when we over estimate t_0, we subsequently over estimate $N(0)$, and vice-versa. In fact, the histogram of the posterior distribution for $N(0)$ also shows positive skewness.

Consider a simulation where $t_0 > 1000$, and hence the population crash has happened before we sample the 50 sequences at 1000 generations bp. When we retrieve the 50 sequences from the population of size 1500, we need only sample a few rare alleles by chance, and we will have an artificially inflated estimate of the genetic diversity (and hence population size). If this inflated estimate of genetic diversity makes it appear as though our population size is close to 15000, then we may retain this simulation as 'similar enough to' ObsDat. This sampling bias is unlikely, but not impossible, and is certainly more likely the larger we make $N(0)$ (as more rare alleles are available via mutation and variation). This makes sense, and we

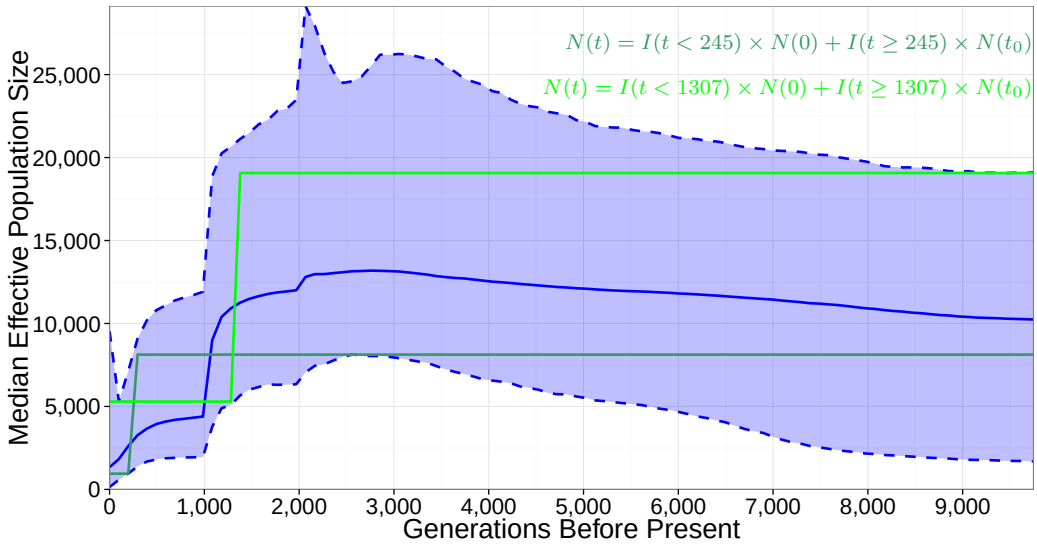


Figure 6.12: Upper and lower estimates (the green lines) for a fitted Bottleneck Model from the BEAST analysis. The blue solid line and shaded area are the posterior sample mean and the corresponding 95% probability interval for $N_e(t)$.

believe that this is what our posterior distribution for t_0 is describing.

To attempt to compare possible values for t_0, we fit a bottleneck population line of the form

$$N(t) = I(t < t_0) \times N(0) + I(t \geq t_0) \times N(t_0)$$

such that the estimated N_e line does not lie outside of the 95% probability limits for the BEAST analysis, and were able to find upper and lower limits for t_0 of (245,1307) (see Figure 6.12).

While these maximum and minimum values of t_0 are similar to the 95% probability interval limits from our analysis, it is of some importance to look at the behaviour of the BEAST analysis at an interval around 600 generations bp. For the BEAST analysis, if we look at the effective population size estimates within 500 and 700 generations before present, the effective population size is no less than 1865, and

no more than 11282 breeding individuals. That is, the 95% probability interval from BEAST does not even contain the true effective population size around the true event time (and this is true from 393 to 984 generations bp). Instead, the BEAST analysis indicates a sudden population decline at 1083 generations bp, and then a more gentle decline until 0 generations bp.

Recall that the prior distribution for the event time was $t_0 \sim U(50, 2000)$, and so our posterior probability interval for t_0 is almost half the width of the prior interval. Importantly though, our analysis is telling us that the population crash probably happened somewhere between the modern sampling time, and the sampling time 1000 generations bp. While this result is encouraging, and our point estimate of 581 is very close to the true parameter value, we investigate whether or not this 95% probability interval width is unusually wide.

Imagine that we had returned median values from the analysis of ABCDat_1 of $M_{N(0)} = 1500$ and $M_{N(t_0)} = 15,000$, and that we use these point estimates in lieu of the intervals

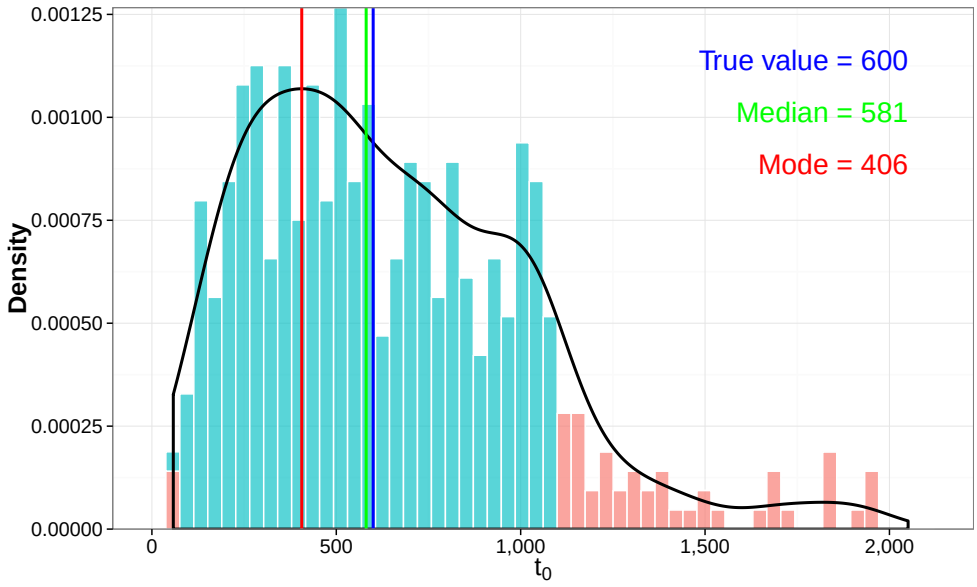
$$N(0) \sim U(M_{N(0)} - s_{N(0)}, M_{N(0)} + s_{N(0)})$$

and

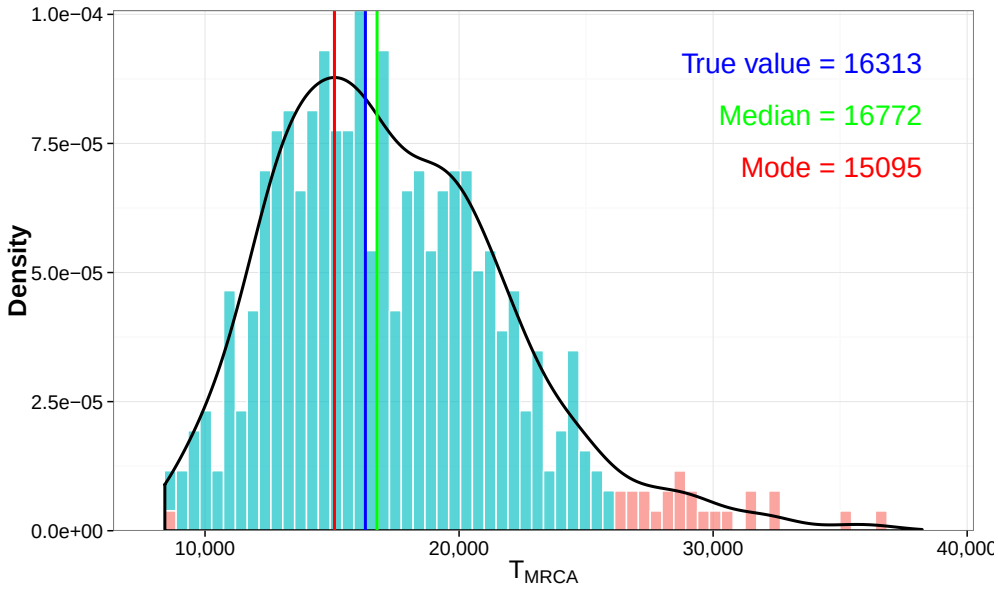
$$N(t_0) \sim U(M_{N(t_0)} - s_{N(t_0)}, M_{N(t_0)} + s_{N(t_0)}),$$

when simulating ABCDat_2. That is, we have perfect estimates for two of our three parameters, and now we must only investigate t_0 using our second ABC step.

Now, for the ideal ABCDat_2, we simulated $150,000$ realisations of the Bottleneck Model with a prior distribution for $N(0) = 1500$, $t_0 \sim U(50, 2000)$ and $\beta^{-1} = 10$ and retained the 500 'closest' simulations for the untransformed posterior estimates. Using this ideal ABCDat_2 we return a posterior median estimate for t_0 of 578 generations bp, and a 95% probability interval of $(50, 1594)$. Clearly, this probability interval is worse than the interval we obtained from our analysis.



(a)



(b)

Figure 6.13: Posterior samples for (a) the event time t_0 and (b) the T_{MRCA} for the Bottleneck Population Model with the true value, the posterior mode of the kernel density estimate and the posterior median indicated. The 95% probability interval is highlighted light blue. The kernel density estimate is given in black.

For completeness we present the annotated tree from the BEAST analysis (see Figure 6.14).

On the Presentation of Results

We presented Figure 6.11 as a comparison for our results with the results recovered from BEAST. Recall from Section 3.1.3 that the BSP BEAST produces is a 95% probability interval for the posterior genealogies, and not a representation of any specific model parameters. Due to the 'model-free' nature of a BEAST analysis, and the fact that BEAST only fits one parameter (the effective population size), this is the most natural way to present a BEAST BSP analysis visually.

When inspecting a BSP, one might assume that any 'path' (series of population size estimates) that remains within the 95% probability interval might be in some sense credible. For example, for our analysis it is tempting to claim that we might be able to infer two population crashes through time (see the blue line in Figure 6.15), since we can place a line within the 95% probability interval. By that same logic, we can infer that the population size might also have undergone a sinusoidal fluctuation, followed by a linear decrease, before a constant size (see the green line in Figure 6.15). Clearly, the green population estimate is nonsense, and in some sense, so is the estimate with two population crashes. This is the case with the BEAST analysis. We may wish to claim a single bottleneck event has occurred, but we must concede that our analysis suggests two periods of population decline occurring either side of the true event time.

The 95% probability interval implies that at any plot time, 95% of the population size estimates for a set of sequences with similar summary statistics to the ObsDat will lie within the corresponding interval. (Recall, a 'plot time' is the discrete times BEAST chooses to estimate population sizes). It is not true that all imagined paths (population dynamic through these estimates) are equally credible, and the more the path deviates from the shape of the median path, the less credible the path,

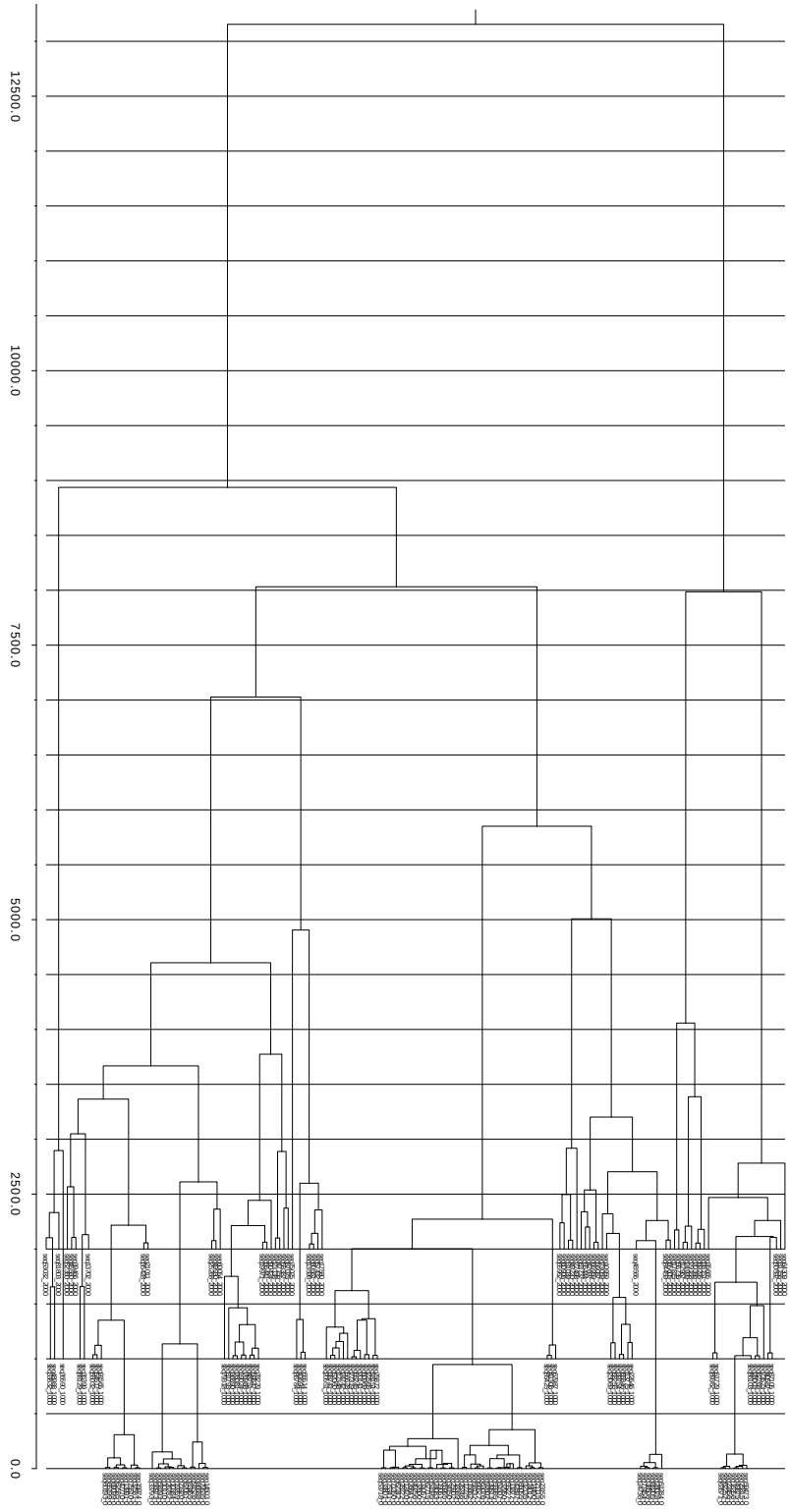


Figure 6.14: The annotated tree produced by the BEAST analysis of our Constant Model data.

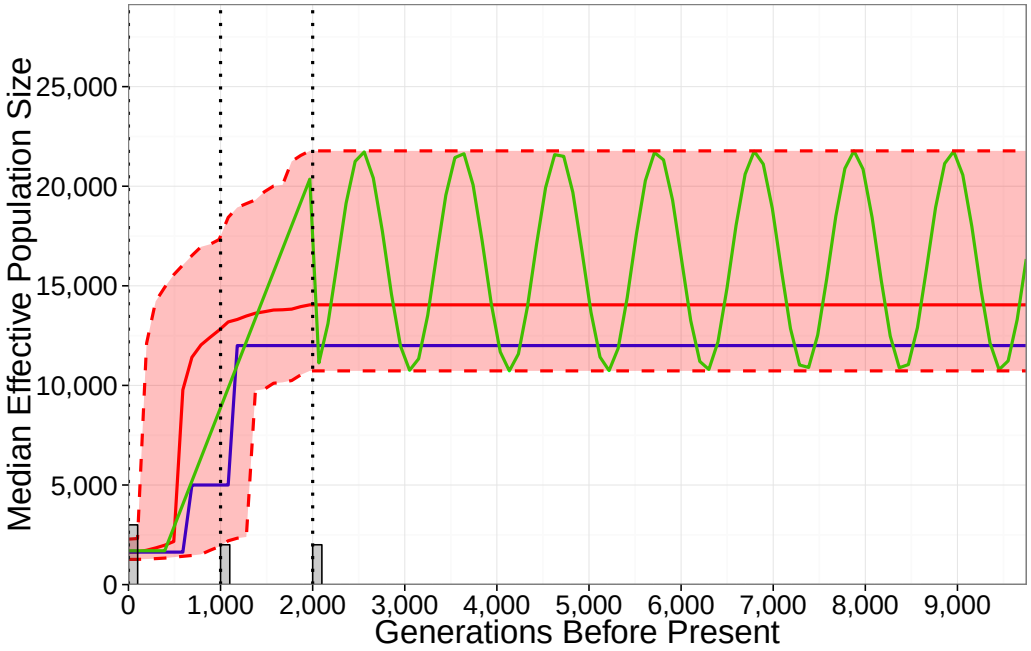


Figure 6.15: The 95% probability intervals for the effective population size via our analysis (the red region), with examples of dubious inferred population dynamics (the blue and green lines), and the recovered median estimates via our method (the solid red line).

given the data.

In the context of our analysis, we made inferences about population model parameters. Specifically, we look at the posterior distributions for $N(0)$, $N(t_0)$ and t_0, and consider 95% probability intervals for these parameters. Visually, $N(0)$ and $N(t_0)$ contribute to variation in the 'y-direction', whereas t_0 contributes to variation in the 'x-direction'. We suggest presenting plotted dynamics with variation only shown in the population size estimate, and accompanying posterior distributions given for any timing parameters (here, t_0 and T_{MRCA}). (Note that BEAST does this by not presenting any of the variation associated with the T_{MRCA} within the BSP, but gives the posterior distribution separately). We present the results of our analysis visually in Figure 6.16.

6.3.1 Bottleneck ObsDat Analysis Conclusions

When we performed model selection for the bottleneck ObsDat, the MLR classification model soft classified the data as Bottleneck Model data with probability 0.9999. When we inspected the PCA plot of the data with the combined TrainDat, it appeared as though the data was closer to the Exponential Model cluster, and specifically the Exponential Model centroid. However, when an analysis of the normalised orthogonally projected distances of the ObsDat from centroid was performed, we found that ObsDat was relatively closer to the Bottleneck Model centroid ($\psi_{Bottleneck} = 0.1053$) than the Exponential Model centroid ($\psi_{Exponential} = 0.1707$). Hence, we correctly classified our data as Bottleneck data, and continued on to the parameter estimation.

We performed our parameter estimation step, and compared the results to the corresponding analysis from BEAST. We immediately noticed that the BEAST analysis was again driven by the sampling times, and this caused the inferred median population size dynamics to display two separate periods of population decline. Our analysis followed the correct bottleneck population model (due to

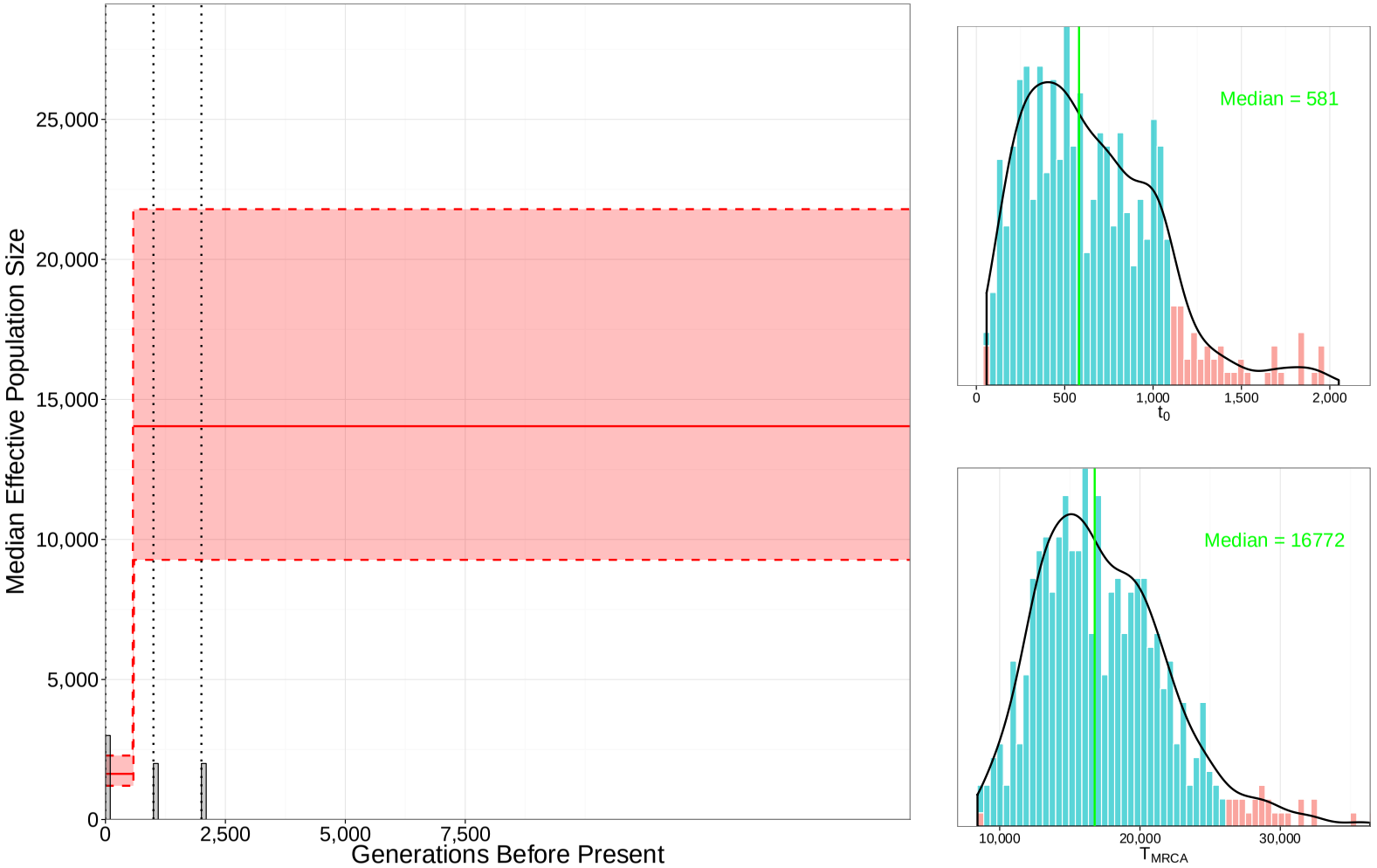


Figure 6.16: The 95% probability intervals for the effective population size (the red region), and the recovered median estimates via our method (the solid red line) via our analysis, with posterior distributions for t_0 and $T_{\mathbf{MRCA}}$.

the fact that we performed a model selection analysis on the ObsDat), and we obtained comparable interval widths to the corresponding BEAST analysis (see Figure 6.11). Importantly though, the 95% probability interval from the BEAST analysis failed to contain the true population size between 393 and 984 generations bp. Our analysis had no such problem.

We pointed out that due to the method BEAST uses to sample a genealogy (and hence a BSP) sequentially via MCMC, the method of presenting these results as vertical 95% probability intervals at equidistant times was sensible, as no other variability can be presented (except possibly for the T_{MRCA}). Since we were estimating population parameters, we do not believe that presenting our results in a similar way was the most informative method possible, as it does not accurately represent the posterior distributions of the parameters of interest.

Instead, we suggest presenting the effective population sizes as intervals with 95% probability intervals according to the model selection decision. We then include posterior distributions for any timing parameters in an effort to reduce confusion about which possible 'paths' a population dynamic could credibly take.

Chapter 7

Conclusions

7.1 Summary

In this thesis we have investigated the problem of recovering population dynamics from a sample of DNA sequences. Early methods for answering this question were limited by the computational power and sequencing methods of the time. A thorough review of these methods highlighted several interesting features of Skyline Plot methodologies.

First, one of the key assumptions for a BSP analysis is panmixia; that every individual in the population has an equal probability of being the child of every individual in the previous generation. This assumption is rarely met, and this bias inflates any population size estimate we might make. Second, the variances around any coalescent event, and specifically the T_{MRCA}, are large, and this variation is not represented in the produced BSP. Similarly, due to the method by which BEAST finds 95% probability intervals at each time that it plots a population size estimate, population events appear to have variation about the time axis. This is certainly not the case, and spurious inferences may be made about the data. Third, BSPs can be largely driven by sampling events. If one can find samples

that are uniformly distributed through time, then bias can be avoided. However, researchers should be aware of this bias.

We proposed a two step process: data driven model selection using multinomial logistic regression (MLR) followed by parameter estimation via Semi-Automatic Approximate Bayesian Computation. Both steps are examples of supervised learning, and this was natural due to the computational efficiency with which we can obtain coalescent simulations. We performed this two step process on observed data (ObsDat) simulated under three models of population dynamics; the Constant, Exponential and Migration Models.

Our method of model selection takes a training data set for each of the candidate models, and fits a MLR model where the model is the categorical response variable, and the insufficient summary statistics are the predictor variables. We can then find a predicted probability of the ObsDat being a product of each of the candidate models. Next, we perform a principal component analysis on the ObsDat and evaluate the normalised, orthogonally projected distances between the ObsDat and the cluster centroids. We do this to identify when model classifications are spurious, or when the correct model is not included in the candidate model set. Both the MLR model and PCA use linear combinations to obtain information about data, and so we plot the first two principal components for the ObsDat to visualise the data.

We compared our model selection method to a common method of model comparison called Bayes Factors. Bayes Factors performed extremely poorly for each ObsDat, and we showed that our method correctly identified the model in each case. We also showed that the summary statistics based on the distances between cluster centroids and the ObsDat were effective in indicating when a MLR classification is spurious. We showed that these distances also performed well when the correct population model was omitted.

Our method of parameter estimation employs Approximate Bayesian Computation

in which we compare semi-automatic summary statistics for the data of interest and a coalescent simulations. These semi-automatic summary statistics are defined by a linear model fit to a training data set, where the insufficient summary statistics are the predictor variables, and the parameters of interest are the response variable (transformed under a Box-Cox transformation).

For each ObsDat we recovered 95% confidence intervals for the population size that contained the true population sizes, and were considerably narrower than the corresponding intervals obtained from the BEAST analysis. Importantly, when the assumption of panmixia was violated by the Migration Model, the 95% probability interval obtained from BEAST did not contain the true population size; an issue our method did not suffer from.

We obtained forward simulated data with a three-dimensional parameter space under a different simulation method called 'forward simulation'. This data was produced under the Bottleneck Model, and hence two of our parameters of interest were concerned with population size estimation, and the third parameter was a timing parameter. The MLR classification correctly identified the population model for the Bottleneck ObsDat, and this finding was supported by the PCA distances. The 95% probability intervals for the two population size parameters both contained the true values, and were of comparable interval width to the associated BEAST analyses. Importantly, the BEAST 95% probability intervals did not contain the true population sizes at all times again, while our estimates did.

The median posterior event time was very close to the true event time, and so we recovered a sensible point estimate for the timing parameter. However, the 95% probability interval for the event time was reasonably wide, but we noticed that the posterior distribution was right-skewed. This indicated that while it was not impossible for a population decline to have happened earlier in our population's history, it was unlikely. We found that the method by which BEAST combines the large sample of posterior genealogies to produce a single plot was not a sensible

plotting method for our parameter estimation scheme. Since we had performed a model selection step, and were investigating specific population parameters, we suggested a different method of plotting results that presented the posterior distributions for timing events, and plotting the point estimate for that parameter instead. In doing so we remove the temptation to fit population dynamics of a dubious nature.

Finally, we consider the purpose of this thesis to be twofold. Current methods suffer from several biases, and researchers can be unaware of the impact these biases have on population size, and event time inferences. However, our method has shown that information contained within the sequence data can be used to identify models of population dynamics and their parameters. Importantly, we have described a method for investigating and visualising *how well* a population model fits the data (a feature Skyline Plots do not have).

7.2 Future Work

Methods that aim to identify past events, such as bottleneck events, must utilise samples from before and after the event occurred. Ancient DNA suffers from degradation, and hence methods involving ancient samples must be able to handle missing data. As discussed in Chapter 6, we have not developed methods for dealing with missing data, and so this is an obvious starting point for future research.

Our methods have relied on code written in the `R Statistical Software`, and coalescent simulations obtained from the command line interface software `BayeSSC`. For this method to be attractive to researchers, a GUI needs to be produced for the entire method.

Appendix A

Appendix

A.1 Basic Terminology

Allele: A unique sequence of characters.

DNA: Sequences made up of the four bases pairs; Adenine, Cytosine, Guanine and Thymine (A,C,G and T).

Genetic Drift: Random fluctuations in the frequency of the appearance of a gene in a small isolated population, owing to sampling variation [1].

Effective Population Size: The total number of individuals capable of producing offspring.

mtDNA: The DNA located in organelles (specialised sub-units within cells) that, in most species, is inherited solely from the mother [17].

Mutation: A permanent change in the DNA sequence of an individual.

Nuclear DNA: The DNA of the chromosomes found in the nucleus of a eukaryotic cell [1].

Recombination: The process that creates new combinations of genes by shuffling the linear order of the DNA [1].

Sequence: A succession of characters describing a part of the genetic makeup of an organism.

A.2 The Backwards Step Algorithm

We employ a backward step algorithm for the goodness of fit for our transformed linear models in Section 4.2. The algorithm seeks to find the sub-model which minimises the Akaike Information Criterion (AIC), by heuristically removing predictor variables one at a time.

We use the general form of the AIC,

$$AIC = 2k - 2\ln(L),$$

where k is the number of parameters in our model, and L is the maximised value of the likelihood function for the model [2].

Let $\mathbf{p} = \{p_1, ..., p_k\}$ be the set of candidate predictor variables for our linear model, $AIC(\mathbf{p})$ be the AIC for M^*, the full linear model with all $p_j \in \mathbf{p}$ (and no interaction terms), and

$$\mathbf{AIC}'(\mathbf{p}) = \{AIC\,(\mathbf{p} \setminus \{p_j\}) \mid\ p_j \in \mathbf{p}\}\,.$$

The algorithm is defined as follows;

Algorithm 6: ABC using Constructed Summary Statistics

Input: $\mathbf{p}_0$, the set of all possible predictor variables.

Set $\mathbf{p}^* = \mathbf{p}_0$;

while $AIC\left(\mathbf{p}^*\right) > min\left(\mathbf{AIC}'(\mathbf{p}^*)\right)$ **do**

  Find i such that $AIC\left(\mathbf{p}^* \setminus \{p_i\}\right) = \min\left(\mathbf{AIC}'(p^*)\right)$;

  $\mathbf{p}^* = \mathbf{p}^* \setminus \{p_i\}$;

end

After performing Algorithm 6, we retain p^* as the 'best' set of predictor variables.

A.3 Fitted Linear Model for the Constant Model Parameter Estimation

For the model

$$\hat{N_e(0)}^{\lambda_{M_1}} = \beta_0 + \sum_{i=1}^{24} \beta_i s_i,$$

we have the following retained summary statistics, s_i, with associated coefficients, β_i, and p-values, p_i.

i	s_i	β_i	p_i	i	s_i	β_i	p_i
0		2.194	$< 2 \times 10^{-16}$	13	π_1	-1.206×10^{-1}	1.54×10^{-5}
1	h_0	6.552×10^{-3}	$< 2 \times 10^{-16}$	14	d_1	1.759×10^{-3}	9.478×10^{-3}
2	H_0	4.780×10^{-1}	1.62×10^{-13}	15	$p_{(1,2)}$	1.236×10^{-3}	$< 2 \times 10^{-16}$
3	d_0	-3.144×10^{-3}	7.04×10^{-10}	16	$\bar{H}_S^{(1,2)}$	8.533×10^{-1}	$< 2 \times 10^{-16}$
4	$p_{(0,1)}$	-4.829×10^{-4}	7.35×10^{-5}	17	$H_T^{(1,2)}$	-4.460×10^{-1}	1.32×10^{-10}
5	$p_{(1,0)}$	1.594×10^{-3}	$< 2 \times 10^{-16}$	18	$F_{ST}^{(1,2)}$	-1.976×10^{-2}	3.51×10^{-4}
6	$H_t^{(0,1)}$	-6.999×10^{-1}	$< 2 \times 10^{-16}$	19	S_2	-2.203×10^{-4}	$< 2 \times 10^{-16}$
7	$F_{ST}^{(0,1)}$	-2.755×10^{-2}	1.76×10^{-6}	20	π_2	-5.193×10^{-2}	4.5450×10^{-2}
8	$p_{(0,2)}$	-3.297×10^{-3}	$< 2 \times 10^{-16}$	21	d_2	-2.693×10^{-3}	5.57×10^{-5}
9	$p_{(2,0)}$	2.801×10^{-3}	$< 2 \times 10^{-16}$	22	$S_{\{0,1,2\}}$	7.173×10^{-4}	$< 2 \times 10^{-16}$
10	$\bar{H}_S^{(0,2)}$	-4.927×10^{-1}	2.09×10^{-9}	23	$\pi_{\{0,1,2\}}$	2.101×10^{-1}	2.86×10^{-7}
11	$F_{ST}^{(0,2)}$	-5.193×10^{-2}	$< 2 \times 10^{-16}$	24	$d_{\{0,1,2\}}$	-4.186×10^{-3}	2.74×10^{-4}
12	S_1	-8.541×10^{-5}	$< 2 \times 10^{-16}$				

A.4 Fitted Linear Models for the Exponential Model Parameter Estimation

For the model (for the initial population size $N_e(0)$)

$$\hat{N_e(0)}^{\lambda^1_{M_2}} = \beta_0 + \sum_{i=1}^{27} \beta_i s_i,$$

we have the following retained summary statistics, s_i, with associated coefficients, β_i, and p-values, p_i.

i	s_i	β_i	p_i	i	s_i	β_i	p_i
0		1.831	$< 2 \times 10^{-16}$	14	S_1	-2.524×10^{-4}	$< 2 \times 10^{-16}$
1	h_0	1.225×10^{-2}	$< 2 \times 10^{-16}$	15	H_1	4.054×10^{-1}	$< 2 \times 10^{-16}$
2	S_0	1.629×10^{-4}	$< 2 \times 10^{-16}$	17	π_1	-1.177	1.16×10^{-6}
3	π_0	-2.518	1.73×10^{-13}	18	d_1	-4.747×10^{-3}	1.13×10^{-13}
4	d_0	-2.289×10^{-2}	$< 2 \times 10^{-16}$	19	$H_T^{(1,2)}$	-3.151×10^{-1}	$< 2 \times 10^{-16}$
5	$p_{(0,1)}$	-2.868×10^{-3}	$< 2 \times 10^{-16}$	20	$F_{ST}^{(1,2)}$	-1.254×10^{-2}	2.59×10^{-3}
6	$p_{(1,0)}$	2.959×10^{-3}	$< 2 \times 10^{-16}$	21	S_2	-4.189×10^{-4}	$< 2 \times 10^{-16}$
7	$\bar{H}_S^{(0,1)}$	-3.846×10^{-1}	$< 2 \times 10^{-16}$	22	π_2	-3.376	$< 2 \times 10^{-16}$
8	$H_T^{(0,1)}$	-8.503×10^{-1}	$< 2 \times 10^{-16}$	23	d_2	6.153×10^{-3}	$< 2 \times 10^{-16}$
9	$F_{ST}^{(0,1)}$	-9.114×10^{-2}	$< 2 \times 10^{-16}$	24	$h_{\{0,1,2\}}$	-6.919×10^{-4}	$< 2 \times 10^{-16}$
10	$p_{(0,2)}$	-4.164×10^{-3}	$< 2 \times 10^{-16}$	25	$S_{\{0,1,2\}}$	3.555×10^{-4}	$< 2 \times 10^{-16}$
11	$p_{(2,0)}$	9.334×10^{-4}	$< 2 \times 10^{-16}$	26	$H_{\{0,1,2\}}$	1.028	$< 2 \times 10^{-16}$
12	$H_T^{(0,2)}$	-1.667×10^{-1}	3.7410^{-11}	27	$\pi_{\{0,1,2\}}$	9.526	$< 2 \times 10^{-16}$
13	$F_{ST}^{(0,2)}$	-9.720×10^{-2}	$< 2 \times 10^{-16}$	24	$d_{\{0,1,2\}}$	-5.865×10^{-2}	$< 2 \times 10^{-16}$

For the model (for the exponential decay parameter K)

$$\hat{K}^{\lambda^2_{M_2}} = \beta_0 + \sum_{i=1}^{27} \beta_i s_i$$

we have the following retained summary statistics, s_i, with associated coefficients, β_i, and p-values, p_i.

i	s_i	β_i	p_i	i	s_i	β_i	p_i
0		6.697×10^{-6}	$< 2 \times 10^{-16}$	14	H_1	-1.640×10^{-5}	$< 2 \times 10^{-16}$
1	h_0	-1.375×10^{-7}	$< 2 \times 10^{-16}$	15	π_1	-4.557×10^{-5}	$< 2 \times 10^{-16}$
2	S_0	-1.436×10^{-8}	$< 2 \times 10^{-16}$	17	d_1	-1.177×10^{-7}	$< 2 \times 10^{-16}$
3	π_0	-2.518	$< 2 \times 10^{-16}$	18	$p_{(1,2)}$	2.251×10^{-8}	4.67×10^{-11}
4	d_0	-4.241×10^{-4}	$< 2 \times 10^{-16}$	19	$\bar{H}_S^{(1,2)}$	1.856×10^{-5}	$< 2 \times 10^{-16}$
5	$p_{(0,1)}$	-2.868×10^{-8}	$< 2 \times 10^{-16}$	20	$H_T^{(1,2)}$	-8.463×10^{-6}	$< 2 \times 10^{-16}$
6	$p_{(1,0)}$	-2.320×10^{-8}	$< 2 \times 10^{-16}$	21	$F_{ST}^{(1,2)}$	-2.067×10^{-6}	$< 2 \times 10^{-16}$
7	$\bar{H}_S^{(0,1)}$	2.177×10^{-5}	$< 2 \times 10^{-16}$	22	S_2	-1.115×10^{-9}	6.14×10^{-4}
8	$F_{ST}^{(0,1)}$	-5.320×10^{-7}	1.98×10^{-4}	23	π_2	3.376×10^{-5}	$< 2 \times 10^{-16}$
9	$p_{(0,2)}$	8.784×10^{-9}	1.040×10^{-3}	24	d_2	-9.019×10^{-7}	$< 2 \times 10^{-16}$
10	$p_{(2,0)}$	6.371×10^{-8}	$< 2 \times 10^{-16}$	25	$h_{\{0,1,2\}}$	-1.939×10^{-8}	1.79×10^{-7}
11	$H_T^{(0,2)}$	-1.686×10^{-5}	$< 2 \times 10^{-16}$	26	$S_{\{0,1,2\}}$	3.670×10^{-8}	$< 2 \times 10^{-16}$
12	$F_{ST}^{(0,2)}$	-1.184×10^{-6}	$< 2 \times 10^{-16}$	27	$H_{\{0,1,2\}}$	1.020×10^{-5}	7.22×10^{-10}
13	S_1	-6.013×10^{-9}	$< 2 \times 10^{-16}$	24	$d_{\{0,1,2\}}$	6.117×10^{-6}	$< 2 \times 10^{-16}$

A.5 Fitted Linear Models for the Migration Model Parameter Estimation

For the model

$$\hat{N_e(0)}^{\lambda_{M_3}} = \beta_0 + \sum_{i=1}^{20} \beta_i s_i,$$

we have the following retained summary statistics, s_i, with associated coefficients, β_i, and p-values, p_i.

i	s_i	β_i	p_i	i	s_i	β_i	p_i
0		1.785	$< 2 \times 10^{-16}$	10	S_1	-6.142×10^{-4}	1.28×10^{-6}
1	h_0	1.779×10^{-2}	$< 2 \times 10^{-16}$	11	π_1	5.812	$< 2 \times 10^{-16}$
2	S_0	-4.738×10^{-4}	8.85×10^{-5}	12	d_2	-6.682×10^{-2}	1.77×10^{-5}
3	π_0	7.706	$< 2 \times 10^{-16}$	13	$p_{(1,2)}$	2.375×10^{-3}	$< 2 \times 10^{-16}$
4	d_0	-9.213×10^{-2}	5.82×10^{-10}	14	S_2	-7.058×10^{-4}	5.23×10^{-8}
5	$p_{(0,1)}$	-2.091×10^{-3}	1.31×10^{-10}	15	π_2	6.084	$< 2 \times 10^{-16}$
6	$p_{(1,0)}$	3.996×10^{-3}	$< 2 \times 10^{-16}$	16	d_2	-3.933×10^{-2}	1.42×10^{-2}
7	$\bar{H}_S^{(0,1)}$	4.856×10^{-1}	5.40×10^{-5}	17	$S_{\{0,1,2\}}$	2.712×10^{-3}	$< 2 \times 10^{-16}$
8	$p_{(0,2)}$	-9.151×10^{-3}	$< 2 \times 10^{-16}$	18	$\pi_{\{0,1,2\}}$	-1.567×10^{1}	$< 2 \times 10^{-16}$
9	$p_{(2,0)}$	7.092×10^{-3}	$< 2 \times 10^{-16}$	19	$d_{\{0,1,2\}}$	-6.780×10^{-2}	3.03×10^{-5}

A.6 Fitted Linear Models for the Bottleneck Model Parameter Estimation

For the model (for the bottleneck size $N(0)$)

$$\hat{N(0)}^{\lambda^1_{M_4}} = \beta_0 + \sum_{i=1}^{30} \beta_i s_i,$$

we have the following retained summary statistics, s_i, with associated coefficients, β_i, and p-values, p_i.

i	s_i	β_i	p_i	i	s_i	β_i	p_i
0		5.543912	$< 2 \times 10^{-16}$	16	S_1	-7.801×10^{-4}	4.11×10^{-6}
1	h_0	4.748×10^{-1}	$< 2 \times 10^{-16}$	17	π_1	-1.787	7.32510^{-3}
2	S_0	-7.179×10^{-3}	$< 2 \times 10^{-16}$	18	d_1	-1.893×10^{-1}	$< 2 \times 10^{-16}$
3	H_0	-3.390×10^{4}	2.09×10^{-5}	19	$p_{(1,2)}$	-1.893×10^{-1}	$< 2 \times 10^{-16}$
4	π_0	5.513	4.59×10^{-16}	20	$\bar{H}_S^{(1,2)}$	-2.977×10^{4}	4.47×10^{-4}
5	d_0	-5.151×10^{-1}	$< 2 \times 10^{-16}$	21	$H_T^{(1,2)}$	-7.334×10^{3}	7.9858×10^{-2}
6	$p_{(0,1)}$	-4.539×10^{-1}	$< 2 \times 10^{-16}$	22	$F_{ST}^{(1,2)}$	-1.558×10^{-1}	6.9479×10^{-2}
7	$p_{(1,0)}$	3.155×10^{-2}	$< 2 \times 10^{-16}$	23	S_2	2.589×10^{-3}	$< 2 \times 10^{-16}$
8	$\bar{H}_S^{(0,1)}$	3.528×10^{4}	7.91×10^{-6}	24	π_2	-5.046	1.78×10^{-13}
9	$H_T^{(0,1)}$	-1.100×10^{4}	7.9853×10^{-2}	25	d_2	-2.698×10^{-1}	$< 2 \times 10^{-16}$
10	$F_{ST}^{(0,1)}$	-1.610	$< 2 \times 10^{-16}$	26	$h_{\{0,1,2\}}$	1.358×10^{-1}	$< 2 \times 10^{-16}$
11	$p_{(0,2)}$	7.612×10^{-2}	$< 2 \times 10^{-16}$	27	$S_{\{0,1,2\}}$	4.873×10^{-3}	$< 2 \times 10^{-16}$
12	$p_{(2,0)}$	-1.576×10^{-1}	$< 2 \times 10^{-16}$	28	$H_{\{0,1,2\}}$	2.246×10^{4}	7.9901×10^{-2}
13	$\bar{H}_S^{(0,2)}$	3.527×10^{4}	$< 2 \times 10^{-16}$	29	$\pi_{\{0,1,2\}}$	1.219×10^{1}	2.62×10^{-14}
14	$H_T^{(0,2)}$	-1.099×10^{4}	8.0013×10^{-2}	30	$d_{\{0,1,2\}}$	6.619×10^{-1}	$< 2 \times 10^{-16}$
15	$F_{ST}^{(0,2)}$	-3.215	$< 2 \times 10^{-16}$				

For the model (for the event time t_0)

$$\hat{t_0}^{\lambda^2_{M_4}} = \beta_0 + \sum_{i=1}^{28} \beta_i s_i,$$

we have the following retained summary statistics, s_i, with associated coefficients, β_i, and p-values, p_i.

i	s_i	β_i	p_i	i	s_i	β_i	p_i
0		4.889×10^2	$< 2 \times 10^{-16}$	15	H_1	1.480×10^6	2.77×10^{-4}
1	h_0	-1.384×10^1	$< 2 \times 10^{-16}$	16	π_1	-1.091×10^2	4.6210^{-6}
2	S_0	-4.665×10^{-1}	$< 2 \times 10^{-16}$	17	d_1	-2.898	1.12×10^{-8}
3	H_0	-1.479×10^6	2.78×10^{-4}	18	$p_{(1,2)}$	-1.653×10^1	$< 2 \times 10^{-16}$
4	π_0	3.195×10^2	$< 2 \times 10^{-16}$	19	$\bar{H}_S^{(1,2)}$	-2.959×10^6	2.77×10^{-4}
5	d_0	-1.988×10^1	$< 2 \times 10^{-16}$	20	$H_T^{(1,2)}$	-1.939×10^2	2.43×10^{-12}
6	$p_{(0,1)}$	-1.828×10^1	$< 2 \times 10^{-16}$	21	$F_{ST}^{(1,2)}$	5.177×10^1	$< 2 \times 10^{-16}$
7	$p_{(1,0)}$	-5.681	$< 2 \times 10^{-16}$	22	S_2	5.165×10^{-1}	$< 2 \times 10^{-16}$
8	$H_T^{(0,1)}$	-2.736×10^2	$< 2 \times 10^{-16}$	23	π_2	-3.968×10^2	$< 2 \times 10^{-16}$
9	$F_{ST}^{(0,1)}$	-2.522×10^1	1.54×10^{-10}	24	d_2	-1.707	3.780×10^{-3}
10	$p_{(0,2)}$	1.980×10^1	$< 2 \times 10^{-16}$	25	$h_{\{0,1,2\}}$	1.451×10^1	$< 2 \times 10^{-16}$
11	$p_{(2,0)}$	-2.376×10^1	$< 2 \times 10^{-16}$	26	$S_{\{0,1,2\}}$	7.290×10^{-2}	6.85×10^{-6}
12	$\bar{H}_S^{(0,2)}$	2.959×10^6	2.78×10^{-4}	27	$H_{\{0,1,2\}}$	3.035×10^2	5.15×10^{-11}
13	$F_{ST}^{(0,2)}$	-1.226×10^2	$< 2 \times 10^{-16}$	28	$\pi_{\{0,1,2\}}$	4.032×10^1	$< 2 \times 10^{-16}$
14	S_1	2.621×10^{-2}	4.298×10^{-3}				

For the model (for the base size $N(t_0)$)

$$\hat{N(t_0)}^{\lambda^3_{M_4}} = \beta_0 + \sum_{i=1}^{28} \beta_i s_i,$$

we have the following retained summary statistics, s_i, with associated coefficients, β_i, and p-values, p_i.

i	s_i	β_i	p_i	i	s_i	β_i	p_i
0		4.905	$< 2 \times 10^{-16}$	13	π_1	-4.954	$< 2 \times 10^{-16}$
1	h_0	-1.858×10^{1}	$< 2 \times 10^{-16}$	14	d_1	3.664×10^{-2}	3.4110^{-12}
2	H_0	7.314×10^{3}	7.8519×10^{-2}	15	$H_T^{(1,2)}$	1.332	7.32×10^{-10}
3	π_0	-7.817	$< 2 \times 10^{-16}$	16	$F_{ST}^{(1,2)}$	-3.532×10^{-1}	$< 2 \times 10^{-16}$
4	d_0	7.387×10^{-2}	$< 2 \times 10^{-16}$	17	S_2	-6.919×10^{-4}	2.1×10^{-5}
5	$p_{(0,1)}$	8.404×10^{-2}	$< 2 \times 10^{-16}$	18	π_2	-5.407	$< 2 \times 10^{-16}$
6	$p_{(1,0)}$	-1.141×10^{-2}	2.04×10^{-10}	19	d_2	4.574×10^{-2}	8.40×10^{-14}
7	$\bar{H}_S^{(0,1)}$	-1.463×10^{4}	7.8480×10^{-2}	20	$h_{\{0,1,2\}}$	6.781×10^{-3}	2.65×10^{-4}
8	$H_T^{(0,1)}$	1.216	3.71×10^{-5}	21	$S_{\{0,1,2\}}$	1.107×10^{-2}	$< 2 \times 10^{-16}$
9	$p_{(0,2)}$	2.443×10^{-2}	2.10×10^{-14}	22	$H_{\{0,1,2\}}$	-1.091	1.4867×10^{-2}
10	$p_{(2,0)}$	-3.882×10^{-2}	$< 2 \times 10^{-16}$	23	$\pi_{\{0,1,2\}}$	8.791	$< 2 \times 10^{-16}$
11	S_1	-1.262×10^{-3}	$< 2 \times 10^{-16}$	24	$d_{\{0,1,2\}}$	8.758×10^{-2}	6.85×10^{-6}
12	H_1	7.314×10^{3}	7.8540×10^{-2}				

A.7 Fitted MLR for the Combined Constant, Exponential and Migration TrainDat

We denote the Constant Model as category 1, the Exponential Model as category 2, and the Migration Model as category 3.

For the models

$$P\left(Y^k = m \,|\mathbf{X}\right) = \frac{e^{\boldsymbol{\beta}^m \cdot \mathbf{x}^k}}{\sum_{c=1}^{3} e^{\boldsymbol{\beta}^c \cdot \mathbf{x}^k}}$$

we have the following retained summary statistics, x_i^k, with associated coefficients, β_i^m, for $m = 1, 2$ and $i = 1, \cdots, 33$. (Recall $\boldsymbol{\beta}^1 = \mathbf{0}$).

i	x_i^k	β_i^2	β_i^3	i	x_i^k	β_i^2	β_i^3
0		8.909×10^{1}	5.309×10^{1}	17	S_1	1.521×10^{-2}	2.117×10^{-2}
1	h_0	2.056×10^{-1}	-5.604×10^{-2}	18	H_1	2.820	3.818
2	S_0	7.393×10^{-2}	1.262×10^{-2}	19	π_1	-6.511×10^{1}	-1.29510^{1}
3	H_0	-1.484×10^{1}	3.081×10^{1}	20	d_1	1.775	-2.227×10^{-2}
4	π_0	-6.170×10^{1}	1.620	21	$p_{(2,1)}$	-1.675×10^{-1}	3.697×10^{-3}
5	d_0	2.256	-1.623×10^{-1}	22	$\bar{H}_S^{(1,2)}$	4.549	-3.851×10^{1}
6	$p_{(0,1)}$	-1.739×10^{-1}	-3.897×10^{-2}	23	$H_T^{(1,2)}$	-4.612×10^{1}	-1.776×10^{1}
7	$p_{(1,0)}$	-1.430×10^{-1}	-2.533×10^{-2}	24	$F_{ST}^{(1,2)}$	8.604	-2.295×10^{2}
8	$\bar{H}_S^{(0,1)}$	-6.010	1.731×10^{2}	25	h_2	-2.74×10^{-2}	7.815×10^{-3}
9	$H_T^{(0,1)}$	-1.766×10^{-1}	1.78×10^{1}	26	H_2	6.277	-8.085×10^{1}
10	$F_{ST}^{(0,1)}$	-5.302	-2.878×10^{2}	27	π_2	-1.007×10^{2}	-3.375×10^{1}
11	$p_{(0,2)}$	2.750×10^{-2}	1.135×10^{-1}	28	d_2	1.414	1.534×10^{-1}
12	$p_{(2,0)}$	1.540×10^{-1}	7.756×10^{-2}	29	$h_{\{0,1,2\}}$	2.606×10^{-1}	8.237×10^{-2}
13	$\bar{H}_S^{(0,2)}$	-4.280	-2.501×10^{1}	30	$S_{\{0,1,2\}}$	-1.880×10^{-2}	3.739×10^{-3}
14	$H_T^{(0,2)}$	-2.527×10^{1}	9.911	31	$H_{\{0,1,2\}}$	-2.773×10^{1}	1.536×10^{1}
15	$F_{ST}^{(0,2)}$	5.515	-2.370×10^{2}	32	$\pi_{\{0,1,2\}}$	-8.500×10^{1}	-6.612×10^{1}
16	h_1	2.365×10^{-1}	4.240×10^{-2}	33	$d_{\{0,1,2\}}$	-4.882	4.363

A.8 Fitted MLR for the Combined Constant, Exponential, Migration and Bottleneck Train-Dat

We denote the Constant Model as category 1, the Exponential Model as category 2, the Migration Model as category 3 and the Bottleneck Model as category 4.

For the models

$$P\left(Y^k = m \,|\mathbf{X}\right) = \frac{e^{\boldsymbol{\beta}^m \cdot \mathbf{x}^k}}{\sum_{c=1}^{3} e^{\boldsymbol{\beta}^c \cdot \mathbf{x}^k}}$$

we have the following retained summary statistics, x_i^k, with associated coefficients, β_i^m, for $m = 1, 2, 3$ and $i = 1, \cdots, 35$. (Recall $\boldsymbol{\beta}^1 = \mathbf{0}$).

i	x_i^k	β_i^1	β_i^2	β_i^3
0		-3.598×10^1	-1.742×10^1	-8.560×10^1
1	h_0	3.627×10^{-1}	4.513×10^{-1}	4.671×10^{-1}
2	S_0	1.617×10^{-2}	8.734×10^{-2}	-0.007×10^{-3}
3	H_0	1.513×10^1	-1.392	-1.334×10^2
4	π_0	2.601×10^1	-7.144×10^1	-1.800×10^2
5	d_0	-4.605×10^{-2}	2.975	-4.235
6	$p_{(0,1)}$	2.214×10^{-1}	2.967×10^{-2}	2.215×10^{-1}
7	$p_{(1,0)}$	-5.488×10^{-2}	-2.953×10^{-1}	1.334×10^{-2}
8	$\bar{H}_S^{(0,1)}$	1.637×10^1	7.538	-8.672×10^1
9	$H_T^{(0,1)}$	-5.312×10^1	-2.997×10^1	1.654×10^1
10	$F_{ST}^{(0,1)}$	6.141	2.728	2.658×10^2
11	$p_{(0,2)}$	3.656×10^{-2}	-2.645×10^{-1}	7.678×10^{-2}
12	$p_{(2,0)}$	-2.165×10^{-1}	-5.551×10^{-1}	-1.756×10^{-1}
13	$\bar{H}_S^{(0,2)}$	1.304×10^1	-2.297	-3.053×10^1
14	$H_T^{(0,2)}$	-7.265	4.089×10^1	6.088×10^1
15	$F_{ST}^{(0,2)}$	-6.735×10^{-2}	1.090×10^1	-1.103×10^2

i	x_i^k	β_i^1	β_i^2	β_i^3
16	h_1	8.644×10^{-2}	1.263×10^{-1}	2.589×10^{-1}
17	S_1	2.086×10^{-2}	3.472×10^{-2}	1.918×10^{-2}
18	H_1	1.762×10^{1}	1.647×10^{1}	-4.000×10^{1}
19	π_1	-7.341×10^{1}	-2.861×10^{1}	-4.304×10^{1}
20	d_1	1.724	8.769×10^{-1}	3.759×10^{-1}
21	$p_{(1,2)}$	1.906×10^{-1}	-5.659×10^{-2}	1.842×10^{-2}
22	$p_{(2,1)}$	2.138×10^{-1}	-2.217×10^{-2}	1.400×10^{-1}
23	$\bar{H}_S^{(0,1)}$	1.428×10^{1}	6.640	1.619×10^{1}
24	$H_T^{(0,1)}$	2.945×10^{1}	-3.099×10^{1}	1.014×10^{2}
25	$F_{ST}^{(0,1)}$	-2.920×10^{-1}	3.923	-2.064×10^{2}
26	h_2	1.096×10^{-1}	1.607×10^{-1}	2.147×10^{-1}
27	S_2	1.138×10^{-2}	-3.450×10^{-2}	2.495×10^{-2}
28	H_2	1.095×10^{1}	-3.195	7.239×10^{1}
29	π_2	1.160	-9.670×10^{1}	-4.742×10^{1}
30	d_2	3.701×10^{-1}	1.666	4.266×10^{-1}
31	$h_{\{0,1,2\}}$	-2.251×10^{-1}	3.894×10^{-1}	-3.559×10^{-1}
32	$S_{\{0,1,2\}}$	-5.307×10^{-2}	-5.651×10^{-2}	2.225×10^{-2}
33	$H_{\{0,1,2\}}$	-2.438×10^{1}	-6.317	7.526×10^{1}
34	$\pi_{\{0,1,2\}}$	3.665×10^{1}	-6.941×10^{1}	-1.307×10^{2}
35	$d_{\{0,1,2\}}$	-2.569	-6.774	1.212×10^{1}

Bibliography

[1] http://medical-dictionary.thefreedictionary.com.

[2] A. Hirotugu. A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[3] C.P. Barnes, S. Filippi, M.P. Stumpf, and T. Thorne. Considerate Approaches to Constructing Summary Statistics for ABC Model Selection. *Statistics and Computing*, 22(6):1181–1197, 2012.

[4] M.A. Beaumont. Approximate Bayesian Computation in Evolution and Ecology. *Annual Review of Ecology, Evolution, and Systematics*, 41:379–406, 2010.

[5] J. Berger. The Case for Objective Bayesian Analysis. *Bayesian Analysis*, 1(3):385–402, 2006.

[6] G.E.P. Box and D.R. Cox. An Analysis of Transformations. *Journal of the Royal Statistical Society. Series B (Methodological)*, 26(2):pp. 211–252, 1964.

[7] S. Cha. Comprehensive Survey on Distance/Similarity Measures Between Probability Density Functions. *International Journal of Mathematical Models and Methods in Applied Science.*, 1(4):300–307, 2007.

[8] K. Csillery, M.G. Blum, O.E. Gaggiotti, and O. Francois. Approximate Bayesian Computation (ABC) in Practice. *Trends in Ecology and Evolution*, 25(7):410–418, 2010.

[9] A.J. Drummond and A. Rambaut. BEAST: Bayesian Evolutionary Analysis by Sampling Trees. *BMC Evolutionary Biology*, 7(1):214, 2007.

[10] A.J. Drummond, A. Rambaut, B. Shapiro, and O.G. Pybus. Bayesian Coalescent Inference of Past Population Dynamics from Molecular Sequences. *Molecular Boiology and Evolution*, 22(5):1185–1192, May 2005.

[11] M. Escabias, A. Escabias, M. Ana, and M.J. Valderrama. Principal Component Estimation of Functional Logistic Regression: Discussion of Two Different Approaches. *Journal of Nonparametric Statistics*, 16(3-4):365–384, 2004.

[12] L. Excoffier, J. Novembre, and S. Schneider. SIMCOAL: A General Coalescent Program for the Simulation of Molecular Data in Interconnected Populations with Arbitrary Demography. *PubMed*, 91(6):506–509, November 2000.

[13] P. Fearnhead and D. Prangle. Constructing Summary Statistics for Approximate Bayesian Computation: Semi-automatic Approximate Bayesian Computation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(3):419–474, June 2012.

[14] J. Felsenstein. Evolutionary Trees from DNA Sequences: A Maximum Likelihood Approach. *Journal of Molecular Evolution*, 17(6):368–376, 1981.

[15] J. Felsenstein. *Inferring Phylogenies.* Sinauer Associates, 2004.

[16] R.R. Hudson, M. Slatkin, and W.P. Maddison. Estimation of Levels of Gene Flow from DNA Sequence Data. *Genetics*, 132(2):583–589, October 1992.

[17] F. Iborra, H. Kimura, and P. Cook. The Functional Organization of Mitochondrial Genomes in Human Cells. *BMC Biology*, 2(1):9, 2004.

[18] R.E. Kass and A.E. Raftery. Bayes Factors. *Journal of the American Statistical Association*, 90(430):773–795, 1995.

[19] L.F. Keller, K. J. Jeffery, P. Arcese, M.A. Beaumont, W.M. Hochachka, J.N. Smith, and M.W. Bruford. Immigration and the Ephemerality of a Natural Population Bottleneck: Evidence from Molecular Markers. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 268(1474):1387–1394, 2001.

[20] J.F.C. Kingman. The Coalescent. *Stochastic Processes and their Applications*, 13(3):235–248, 1982.

[21] W. L. S. Li and A. J. Drummond. Model Averaging and Bayes Factor Calculation of Relaxed Molecular Clocks in Bayesian Phylogenetics. *Molecular Biology and Evolution*, 29(2):751–761, 2012.

[22] P. MacCullagh and J.A. Nelder. *Generalized Linear Models.*, volume 37. CRC Press, 1989.

[23] A. Mankertz. Molecular Biology of Porcine Circoviruses. *Animal Viruses. Molecular Biology*, pages 355–374, 2008.

[24] P. Marjoram, J. Molitor, V. Plagnol, and S. Tavaré. Markov Chain Monte Carlo Without Likelihoods. *Proceedings of the National Academy of Sciences*, 100(26):15324–15328, 2003.

[25] V.N. Minin, E.W. Bloomquist, and M.A. Suchard. Smooth Skyride through a Rough Skyline: Bayesian Coalescent-Based Inference of Population Dynamics. *Molecular Biology and Evolution*, 25(7):1459–1471, 2008.

[26] M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of Machine Learning.* The MIT Press, 2012.

[27] M. Nei. *Molecular Evolutionary Genetics.* Columbia University Press, 1987.

[28] M. Nei and W.H. Li. Mathematical Model for Studying Genetic Variation in Terms of Restriction Endonucleases. *Proceedings of the National Academy of Sciences*, 76(10):5269–5273, October 1979.

[29] M. Nei and F. Tajima. DNA Polymorphism Detectable by Restriction Endonucleases. *Genetics*, 97(1):145–163, January 1981.

[30] T. Nguyen, M. Anh, S. Klaere, and A. von Haeseler. MISFITS: Evaluating the Goodness of Fit between a Phylogenetic Model and an Alignment. *Molecular Biology and Evolution*, 28(1):143–152, 2011.

[31] B. O'Fallon. TreesimJ: A Flexible, Forward Time Population Genetic Simulator. *Bioinformatics*, 26(17):2200–2201, July 2010.

[32] J. Pellicer, M.F. Fay, and L.J. Leitch. The Largest Eukaryotic Genome of Them All? *Botanical Journal of the Linnean Society*, 164(1):10–15, 2010.

[33] O.G. Pybus, A.Rambaut, and P.H. Harveya. An Integrated Framework for the Inference of Viral Population History From Reconstructed Genealogies. *Genetics*, 155:1429–1437, July 2000.

[34] R Development Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2012.

[35] C. Robert. *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation.* Springer, 2007.

[36] C. Robert, J-M. Cornuet, J-M. Marin, and N.S. Pillai. Lack of Confidence in Approximate Bayesian Computation Model Choice. *Proceedings of the National Academy of Sciences*, 108(37):15112–15117, 2011.

[37] D. B. Rubin. Bayesianly Justifiable and Relevant Frequency Calculations for the Applied Statistician. *The Annals of Statistics*, 12(4):pp. 1151–1172, 1984.

[38] M. Slatkin. Rare Alleles as Indicators of Gene Flow. *Evolution*, 39(1):pp. 53–65, 1985.

[39] K. Strimmer and O.G. Pybus. Exploring the Demographic History of DNA Sequences Using the Generalized Skyline Plot. *Molecular Biology and Evolution*, 18(12):2298–2305, 2001.

[40] F. Tajima. Statistical Method for Testing the Neutral Mutation Hypothesis by DNA Polymorphism. *Genetics*, 123(3):585–95, 1989.

[41] S. Tavare. Some Probabilistic and Statistical Problems in the Analysis of DNA Sequences. *Lecture Notes on Mathematical Modelling in the Life Sciences*, 17:57–86, 1986.

[42] T. Toni and M.P.H. Stumpf. Simulation-Based Model Selection for Dynamical Systems in Systems and Population Biology. *Bioinformatics*, 26(1):104–110, 2010.

[43] T. Toni, D. Welch, N. Strelkowa, A. Ipsen, and M.P.H. Stumpf. Approximate Bayesian Computation Scheme for Parameter Inference and Model Selection in Dynamical Systems. *Journal of The Royal Society Interface*, 6(31):187–202, 2009.

[44] W. Vanpaemel. Prior Sensitivity in Theory Testing: An Apologia for the Bayes Factor. *Journal of Mathematical Psychology*, 54(6):491–498, 2010.

[45] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S.* Springer, New York, fourth edition, 2002.

[46] W.N. Venables and B.D. Ripley. *Modern Applied Statistics With S-PLUS.*, volume 250. Springer-verlag New York, 1994.

[47] G.A. Watterson. On the Number of Segregating Sites in Genetical Models Without Recombination. *Theoretical Population Biology*, 7(2):256–276, 1975.