UNIVERSITY OF
CAMBRIDGE

Cladogram estimation and analyses of phenotypic and phylogeographic data

Author: I.Manolopoulou
Supervisor: Prof. S.P. Brooks

Cambridge University
Statistical Laboratory
January 2006

Preface

This essay contains both a literature review and some original research. Throughout the study, a dataset from La Palma island is used and referred to, which was given to me, along with all the graphs relevant to it, by Dr Brent Emerson from the University of East Anglia to whom I am grateful.

Chapter 2 discusses some methods that have already been developed. Apart from the example which I simulated and used for different methods, the rest is not original material. All the methods are cited accordingly.

Chapter 3 is almost entirely original results. Most of the ideas were discussed with my supervisor or suggested by him and implemented by me. Some of the techniques were motivated by the techniques described in the review chapter or by other literature sources (cited throughout the work). An appendix with genetic definitions and an overview of basic genetic techniques and Markov Chain Monte Carlo methods is included at the end.

This essay largely coincides with my 4th term review. No part of it has been published yet, nor has it been submitted elsewhere for a prize or thesis.

Contents

Abstract

We aim at constructing efficient methods of analysing DNA sequence phenotypic and phylogeographic data. We are given a set of sequences from some individuals, along with measurements from a characteristic trait (phenotypic effect), or from their geography or history (phylogeographic data), and we want to identify which mutations show a significant change in the phenotypic effect, or what geological (e.g. formation of a canyon, explosion of a volcano) or historical events (e.g. colonisation) are associated with clusters of individuals.

Such analyses consist of two main steps: Forming the underlying rooted cladogram (similar to a phylogenetic tree, see Appendix) and then looking for edges (representing mutations) separating the data into significant clusters. A number of problems arise in the process: Missing haplotypes (i.e. uncertainty in the intermediate mutations between two sequences), homoplasy (i.e. a specific amino acid position mutating repeatedly) and recombination (two chromosomes exchanging a chunk of their DNA) are some of the main ones.

We describe some of the main methods, both classical and Bayesian, that have been implemented so far in order to analyse such data efficiently, and their advantages and limitations. We then present Markov Chain Monte Carlo (MCMC) methods of carrying out the cladogram estimation and association analyses simultaneously. Our approach allows for uncertainty over the true tree (and other parameters like amino-acid frequencies, mutation probabilities etc.) to be carried throughout the analyses, and also for the number of significant clusters to vary, using a Reversible Jump Markov Chain Monte Carlo (RJMCMC) algorithm. We use both a simulated dataset and a real dataset from La Palma island for Brachyderes Rugatus Rugatus (Beetles) to discuss the efficiency of the methods presented throughout the study.

Chapter 1

Introduction

This study is motivated by the increasing interest in phenotypic† and phylogeographic† analyses based on recent developments in genetic research [1]. Current statistical tools are often not powerful enough to analyse the data and draw reliable conclusions, so there is a need for new methods of analyses. Some of the main questions include identifying genes† and mutations† associated with characteristic traits or disease (phenotypic analyses), or drawing conclusions about the evolutionary and geographical history of a species (phylogeographic analyses).

These are important for many reasons: in the former case, such analyses may be used to prevent or predict unwanted traits or disease, encourage the development of desired ones and even control specific genes to alter the phenotypic effect. In the latter case, they can help in understanding the long-term evolutionary history and behaviour of species (e.g. colonisation) and investigating the effect of geological events(e.g. formation of a canyon, explosion of a volcano).

These questions represent a practically very similar statistical problem: We are given a set of DNA sequences from some individuals, along with one or more measurements of a characteristic trait, or geographical (i.e. the coordinates where the individual was found) or historical (i.e. when) measurements. We want to find significant clusters of individuals and associate mutations or geological events with the differences.

Our aim is to construct an efficient method of estimating the underlying rooted cladogram† for a set of sequence data and interpreting the corresponding phenotypic data using Markov Chain Monte Carlo (MCMC) methods (for an overview of MCMC methods see Appendix). Due to multiple level uncertainty in the genetic structure of individuals and the discrete dependent parameters, classical methods are not powerful and can lead to false conclusions.

We assume that we have a set of aligned† sequence data and from each haplotype we have measurements of a phenotypic effect from one or more individuals. Our first aim is to construct a network(s) (or cladogram) representing the history of the haplotype† structure, where nodes correspond to sequences, and to make inference about its root, representing the Most Recent Common Ancestor (MRCA) of the haplotypes in our dataset.

Subsequently, we want to associate mutations with significant differences in the phenotypic trait using our cladogram. More than one mutations may appear correlated to the phenotype, but only one (or more) is actually directly related. Hence, once a cladogram has been chosen, one also has to choose a set of edges (representing mutations) which are most significant with respect to the characteristic trait. So we are dealing with a clustering problem, where only specific clusterings are allowed, namely ones formed by separating our data by removing edges from the cladogram.

There is a lot of uncertainty in both deciding what the true rooted cladogram is and which edges show a significant effect. There are 3 main sources of uncertainty in forming the true cladogram: missing nodes, homoplasy† and recombination†. In terms of the significant separating set of edges, another source of uncertainty is the fact that the size of the significant set is unknown.

By **missing nodes** we mean that the set of haplotypes we have in our data does not form a connected graph. Sometime in history, a haplotype mutated again and again, and one of the intermediate ones died out, or are just not in our sample. Reconstructing the history and deciding on what the missing sequences are is not always deterministic (see Figure 1.1).

[1] for a glossary and overview of biological terms marked with †, see Appendix

Homoplasy occurs when a mutation takes place at a certain amino-acid† site more than once. This can lead to the presence of loops in our cladogram, even though in reality a loop didn't occur. Even at homoplasy events, for example a loop of length 4 represents 2 homoplasy events, whereas if we remove one of the edges then we only have 1 homoplasy event but still the same haplotype structure, which is much more likely. Obviously there are many ways in which loops may be removed from our network (see example in Figure 1.2).

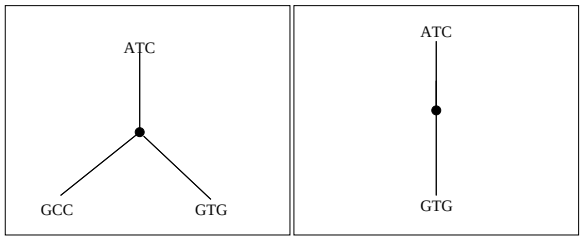


Figure 1.1: An example of a deterministic missing node (left) and a non-deterministic (right). In the case on the left, the missing node necessarily has to be GTC, whereas on the right it could be either GTC or ATG.

Lastly, **recombination** happens when two haplotypes exchange a whole section of their DNA sequence. In our cladogram, an apparent homoplasy event could be the result of recombination, and so could a branch in the cladogram which is disconnected from the rest of the cladogram. Again, there is a very large number of possible recombinations which can lead to the observed set of haplotypes.

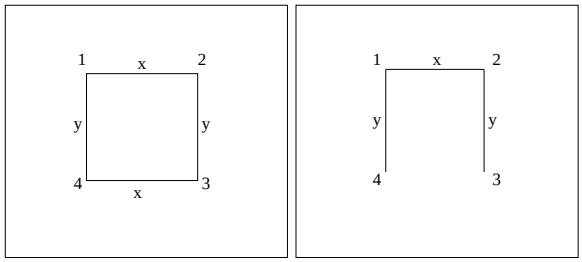


Figure 1.2: An example of two possible underlying histories which result in the same sequence set. Here x and y are two positions on the sequences which mutate. Clearly, the one on the left is a lot less likely than the one on the right. Removing any of the 4 edges would leave us with the same observed sequences.

1.1 Methods of analyses

A lot of methods have been implemented for estimating the true tree resulting to our DNA sequences, some of which are described here (Chapter 2). Most of them are aimed at drawing conclusions about the history of a species and hence the trees formed have branch lengths associated with lengths of time (phylogenetic trees). For many of the analyses here, such a tree is not necessary and so we discuss current methods of estimating the cladogram.

In terms of analysing the phenotypic (or other) data in relation to the tree, one of the main methods available is Nested Clade Analyses (NCA). We describe how it works and present 2 main examples, a simulated dataset (presented below) and the Beetle dataset from La Palma used by Emerson et al. (2005) (see Appendix for details about the dataset).

We then present MCMC methods of performing the steps described above (Chapter 3), and implement it both for the simulated dataset and for the Beetle data.

1.2 Simulation of data

Most real sets of data involve all of the complications described above, so to develop an efficient algorithm we use simulated sets of data so that we can control the presence of missing nodes, homoplasy and recombination. In each species and each locus in the genetic material of that species, different models and assumptions are appropriate. Here we present the general model and then use specific values applicable to the examples we use each time.

We assume that the amino acid frequencies are at equilibrium and that the mutation process is time-reversible. We use a Generalised Time-Homogeneous Time-Reversible Markov Process model

(REV, see Tavaré (1986)) for the mutation rates, assumed independent and identical across all sites, generated by a Q-matrix

$$Q = \begin{pmatrix} \cdot & \alpha\pi_G & \beta\pi_C & \gamma\pi_T \\ \alpha\pi_A & \cdot & \delta\pi_C & \epsilon\pi_T \\ \beta\pi_A & \delta\pi_G & \cdot & \zeta\pi_T \\ \gamma\pi_A & \epsilon\pi_G & \zeta\pi_C & \cdot \end{pmatrix}$$

where the π_i's ($i = A,\ G,\ C,\ T$) represent the equilibrium probabilities of the amino-acids (we assume that the gene frequencies are stationary), and $\alpha, \ldots, \zeta$ the relative transition and transversion probabilities, so that we have 9 degrees of freedom. From now on for simplicity we refer to ($\pi_A,\ \pi_G,\ \pi_C,\ \pi_T$) as ($\pi_1,\ \pi_2,\ \pi_3,\ \pi_4$) respectively. This chain is time-reversible since $\pi_i q_{ij} = \pi_j q_{ji},\quad i, j = 1, \ldots, 4$. Here we assume that the generator matrix is identical and independent across all sites, in the future it may be allowed to vary.

Looking at the jump chain of the above process we obtain the following jump-matrix:

$$P = \begin{pmatrix} 0 & \kappa\alpha\pi_2 & \kappa\beta\pi_3 & \kappa\gamma\pi_4 \\ \lambda\alpha\pi_1 & 0 & \lambda\delta\pi_3 & \lambda\epsilon\pi_4 \\ \mu\beta\pi_1 & \mu\delta\pi_2 & 0 & \mu\zeta\pi_4 \\ \nu\gamma\pi_1 & \nu\epsilon\pi_2 & \nu\zeta\pi_3 & 0 \end{pmatrix}$$

where $\kappa, \lambda, \mu, \nu$ are such that the rows add up to 1.

The algorithm used to simulate the data is described below.

1.2.1 The simple case: No homoplasy, recombination or missing nodes

First we generate an initial sequence of length l using estimates of the gene frequencies.

We then simulate independent Markov Processes using estimates of the mutation probabilities for each of the amino-acid sites and each of the haplotypes already formed (just the initial one to begin with), not allowing any jumps from sites that have already mutated once.

We use the resulting data to form a tree so that each sequence is represented by a node, and 2 sequences are connected if they are exactly 1 letter apart (i.e. differ at exactly 1 letter). In the case of no homoplasy and recombination, the sequence data defines a unique tree when there are no missing nodes, which we form. If the number of mutations K causing a significant effect is known, in order to simulate a phenotypic effect which is continuous and can assumed to be normal, we then randomly pick K edges, and we assume that the data from the subgraphs formed come from normal distributions with different means but the same variance. At each node, we pick the number of data points by a $\sim$Poisson(λ), say, and generate data points from the corresponding normal distribution. In fact, we take the number of data points to be distributed as $\sim$1+Poisson(λ) to ensure we don't get any nodes with zero data points, since that case would correspond to an unobserved node and would not be in the initial sequence dataset.

If the number of significant separating edges is not fixed, we first pick a number K from some distribution and then continue as above for a fixed K.

Example: Simulated cladogram

Throughout this study we will use both a real and a simulated set of data. For the simulated dataset, we generate a tree of size 40 and simulate a complete dataset with $m = 40$, where there are two significant mutations. Using TCS1.21 for a graphical representation of the tree formed, we obtain Figure 1.3.

1.2.2 Homoplasy and missing nodes

If we want to include homoplasy, then we don't need to make sure that we pick a different amino-acid position every time. In order to obtain a set of data with missing nodes, we simply remove sequences from the set of sequences, so that we can check whether our program really finds the correct missing nodes.

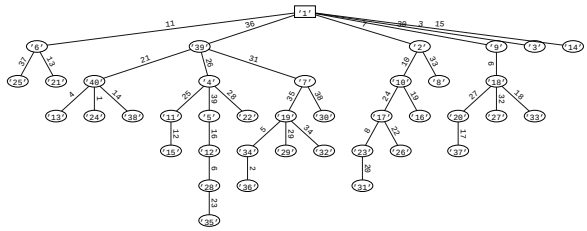


Figure 1.3: Simulated cladogram

1.2.3 Recombination

To simulate a set of data involving recombination, we decide on a (very small) probability p of recombination occurring. Then at each step, with probability $1-p$ we choose to allow a mutation to happen randomly, and with probability p we choose to recombine. To do so, we choose 2 of the existing haplotypes y, z at random, and a number x at random so that $0 < x < l$ (l being the length of the sequences). Then with probability $1/2$ we choose to form a new haplotype with sequence identical to y for all positions $\leq l$ and identical to z in positions $> l$. With probability $1/2$ we do the opposite: copy the left hand side from the z and the right hand side from the y.

Chapter 2

Review of methods

2.1 Introduction

The main method used for phylogeographic analyses of this kind is Nested Clade Analyses. Nested clade analyses are carried out in 3 main steps: forming the cladogram, forming the nested cladogram, and finally testing associations between clades and phenotypic effects. This chapter is structured as follows:

(a) First (Section 2.2) we examine different methods of forming the cladogram, which is in some ways equivalent to forming the phylogenetic tree. There are two main approaches to phylogenetic trees.

The first one is called the *phenetic* approach, and makes no assumptions or inference about the historical relationship between genes and mutations. It makes use of distance measures between individuals and forms a hierarchical clustering type of tree. In other words, it draws no conclusions about which mutations actually happened, instead the clusters are formed based solely on similarity scores between the sequences.

The second approach is the *cladistic* approach, which aims at finding the optimal path of evolution and ancestral relationship between nodes (i.e. individuals/genes) by means of some optimisation algorithm. Here we concentrate on cladistic methods as they give more accurate results, and we try to find the best possible such method.

So far 3 main types of cladistic methods have been developed: maximum likelihood methods, maximum parsimony methods and Bayesian methods. Maximum likelihood methods consider all possible mutational paths and then decide the most likely one to be the true one. On the other hand, maximum parsimony methods assume that the true tree is the one of least total length. Between maximum likelihood and maximum parsimony methods, maximum parsimony is considered by most to be the method of choice because it is the easiest and most practical to implement. Bayesian methods assume a prior distribution on the phylogenetic tree[†] (or cladogram) and derive the best graph(s) based on the evaluation (or estimation) of the posterior distribution. By using MCMC methods, such calculations can be done quite accurately and efficiently. We describe maximum parsimony and various Bayesian methods, and discuss the advantages and disadvantages of both.

(b) Having formed the cladogram or phylogenetic tree, we describe the method in which the nested cladogram is formed (Section 2.3).

(c) We describe the classical way of using the nested cladogram in order to derive conclusions about possible associations between mutations and phenotypic (or phylogeographic) effects (Section 2.4).

2.2 Forming the cladogram

The cladogram represents the evolutionary history of the species or individuals under study. To form such a tree, we need to construct an algorithm which decides what mutational steps led to the haplotype structure observed. Having estimates about the probabilities of different mutations, such inference is possible.

The aim of the cladogram is to identify which mutations actually occurred. For example, if two sequences differ in 2 characters, does that mean that only 2 mutations (substitutions) happened, or is it likely that there exist some other mutations which are not observed? If a restriction site shows evidence of mutating much more frequently than others, then it is likely that there are mutations that are not detectable or are ambiguous. If, for example, a restriction site has mutated $A \to T \to C$, then the maximum parsimony assumption does not hold since two mutations actually took place even though only a one-letter difference is present.

One of the problems of analysing DNA sequence data is the fact that often genes have recombined. In order to construct a valid cladogram, we need to be working on a region of the DNA which hasn't had any recombinations. Recombination makes different parts of the same haplotype represent different evolutionary processes, which would lead to misleading results, and hence we need to exclude them. There is a number of different methods and software o break up the haplotypes into subregions, within which no recombination has occurred, and form a different cladogram for each. In the case of phylogeographic data, using mitochondrial data is very useful since it involves no recombination.

2.2.1 Maximum parsimony methods

Maximum parsimony methods assume that the most likely history of a gene is the one which contains the least number of mutations. A parsimonious relationship between two haplotypes implies that there are no unobserved mutations, i.e. that the only mutations that have occurred correspond to the sites in which the two haplotypes are different, and that these were a result of one mutation each. This may not always be true, which is why testing this assumption is an important part of this method. We proceed in steps, where at step n we calculate the probability that pairs of haplotypes differ by at most n mutations. A more detailed outline of the whole procedure is given below.

In order to test whether the parsimonious assumption is valid, one of the first things we have to do is estimate the maximum parsimony probability of our data. A method is suggested in Templeton et al. (1992), described here, about how to estimate the maximum parsimony probability of our data, and we set our acceptance level by convention at 95%. This means that we will reject our assumption that the number of mutations leading from one haplotype to another one is no more than the observed mutations (i.e. the number of different amino acids) if the probability of our data based on that assumption is less than 5%.

The first thing we do is present a method to construct an estimator for evaluating the limits of parsimony, meaning the smallest (i.e. worst case) probabilities of maximum parsimony being true. Ideally, all sites will be parsimonious, although this is rarely true in reality. In order to estimate the probability that the maximum parsimony assumption is not true, we consider the oldest polymorphic restriction site, the *index site* (but we don't actually try to estimate which one this is). The total probability that two haplotypes differ at the index restriction site, differ at $j-1$ other restriction sites, and share in common the presence of m cut restriction sites (meaning that they have m letters in common in the extracted DNA sequence under consideration, i.e. the sequences we have available) is approximated by:

$$\begin{aligned} L(j,m) &= (1-q_1)\left[1-q_1/b\right](1-q_1)^{2m} \times \left\{2q_1\left[2-q_1(b+1)/b\right]\right\}^{j-1} \\ &\quad \times \left\{1-2q_1\left[1-q_1/b\right]\right\} = \qquad (2.1) \\ &= (2q_1)^{j-1}(1-q_1)^{2m+1}\left[1-q_1/b\right] \times \left[2-q_1(b+1)/b\right]^{j-1} \\ &\quad \times \left\{1-2q_1\left[1-q_1/b\right]\right\}. \qquad (2.2) \end{aligned}$$

Here q_1 is the probability of a nucleotide change in a single site of the two haplotypes since their respective lineages diverged at the index restriction site, m is a constant based on the similarity between the two haplotypes and $b \in [1,3]$ represents the transition bias (compared to transversion),

so that $b = 3$ if there is no bias and $b = 1$ if there is an extreme bias. The value of b is taken to be 3 unless there is evidence to suggest otherwise from previous experiments. A detailed explanation of how the above expression is derived may be found in Templeton et al. (1992).

Combining (2.2) with a uniform prior on q_1, a standard Bayesian estimator of q_1 is thus

$$\hat{q}_1 = \frac{\int_0^1 q_1 L(j,m) dq_1}{\int_0^1 L(j,m) dq_1} \tag{2.3}$$

We now consider mutations that arose after the second oldest mutation associated with a different site. The probability of these mutations in a block of r nucleotides is designated by q_2. An estimator for P_j, the probability that two haplotypes differing at j sites but sharing m have a parsimonious relationship, is:

$$\hat{P}_j = \prod_{i=1}^{j} (1 - \hat{q}_i). \tag{2.4}$$

Having established a formula for estimating the parsimony limits between haplotypes, we iteratively calculate it for pairs of haplotypes starting from 1-step parsimony and moving on to 2-step etc until we have a complete cladogram. In detail, below are the steps we follow:

Step 1: We take $j = 1$ and thus estimate P_1, i.e. the probability of parsimony of haplotype pairs that only differ in one site. If any of them is less than 95%, we terminate the algorithm, as we can't give any accurate results. If not, we link up all haplotypes that differ by one site. In addition, it is often the case that other mutational changes which are usually unique are obvious, and so they can be integrated into our 1-step network (see Lloyd and Calder (1991)).

In this step, we expect to observe homoplasies. However, even at homoplasy events, we should not observe any closed loops. If we do, then this suggests that recombination has occurred, which (although typically would have been detected before starting the cladogram) can be resolved in the next step.

Step 2: The 1-step networks are used to identify potential products of recombination. It has been concluded that recombination should only be inferred if a single recombination event can resolve two or more homoplasies. We first inspect our 1-step networks for homoplasy events, and if they exist, we inspect the haplotypes involved for possible recombination events. A detailed explanation is given in Templeton et al. (1992).

Starting from the ends of the DNA strand being examined, we keep adding sites and constructing maximum parsimony cladograms until there is evidence for no more than one recombination event resulting in sample exclusion. Once these two regions have been identified, we continue on the remaining of the DNA region until it has been completely subdivided into a set of regions in which little or no recombination is believed to have occurred. For each of the subregions, a different cladogram is formed.

Step 3: We now increase j by 1, and calculate P_j for all possible pairs.If parsimony is accepted, we unite the two $(j-1)$-step haplotype networks through the two haplotypes that differ by j steps to form a j-step network.

We repeat step 3 until all haplotypes are in a single connected graph, or in connected subgraphs which between them don't necessarily have a parsimonious relationship. In the case of a high probability of parsimony, we obtain a spanning tree which includes all the observed haplotypes as nodes, and we are done. So far no loops should be present, since we are only looking at sites which are parsimony informative. If we don't have a complete connected graph, we move to step 4.

Step 4: We now unite the separate networks identified in the previous step into a single cladogram, considering both parsimonious and non-parsimonious linkages. Let x be the number of mutational steps involving restriction sites that connect two networks under maximum parsimony. Then, the probability that y or fewer of the x restriction site mutations are non-parsimonious is:

$$\sum_{i=0}^{y} \sum_{I} q_{j(k)} \prod_{k=1}^{x} \left(1 - q_{j(k)}\right) \tag{2.5}$$

where I refers to the set of all permutations of the x age ranks of the mutations. Since we are concerned only with the total number of mutations that occurred beyond those required by parsimony, we need to consider all permutations of the age ranks with which these additional mutations are associated that yield the same number of total additional mutations. This is achieved by placing age ranks into two classes of size i and $x - i$, and then summing over all permutations of the age ranks that result in these class sizes. These alternative permutations are indicated by $j(k)$, which refers to the kth permutation in the set I. The first product in (2.5) is defined to be 1 when $i = 0$.

We then find the minimum value of y such that (2.4) is greater than or equal to 0.95. Our set of plausible cladograms contains all connections between disjoint networks that include the maximum parsimony solutions as well as any connections involving up to y additional mutational steps.

This results to a set of both parsimonious and non-parsimonious networks. In this step, some loops may appear. A number of criteria are used to break them up and conclude one (or more) cladograms to be the true one. These trees are then used to carry out NCA, as described in Section 2.4.

In the case of ambiguity of the haplotype network, when there is no unique most likely tree, various criteria are used. For example, within a cladogram, rare haplotypes are more likely to be tip haplotypes, and common ones more likely to be interior. Also, in the case of phylogeographic data, singleton haplotypes are more likely to be connected to haplotypes from the same population as opposed to haplotypes form different populations.

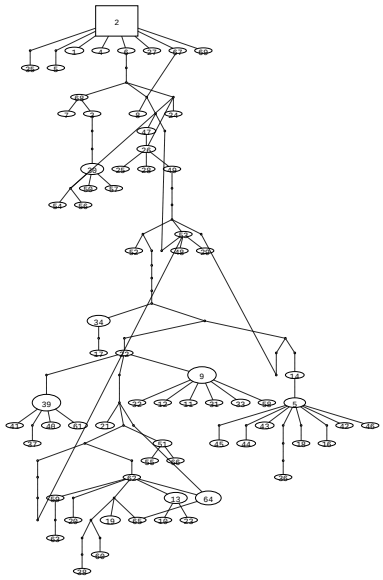


Figure 2.1: Output of TCS 1.21

Example: The beetle data

Using the beetle data from La Palma, Emerson et al. (2005) run TCS 1.21 (a software which forms the cladogram using maximum parsimony methods) to obtain the tree. Because the outcome is not unique, the criteria described above are used to decide on a single one. Using the criteria described in Templeton et al. (1992), and reducing any trivial re-arrangements to one, the remaining cladogram are shown in Figure 2.2. Then network A is assumed to be the true one, since in the other two networks the groups from the north and south are genetically similar which is unlikely.

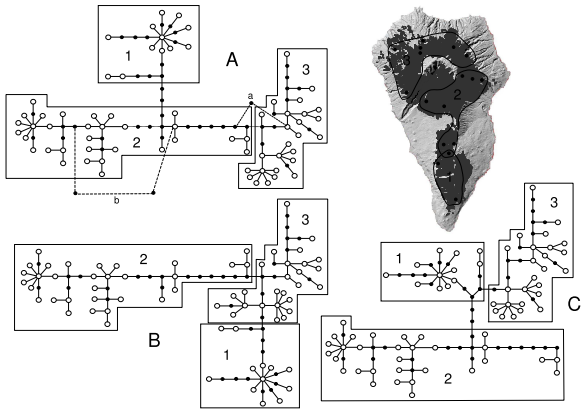


Figure 2.2: The 3 possible cladograms, related to the map

2.2.2 Bayesian Methods

So far the work on Bayesian phenotypic and phylogeographic analyses has been limited. However, a lot of work has been done on Bayesian phylogenetic trees.

Although NCA typically does not require phylogenetic trees, but makes use of a cladogram, a review of MCMC methods on phylogenetics provides us with useful tools and ideas on how to proceed with cladograms and phenotypic/phylogeographic analyses. A phylogenetic tree is similar to a cladogram: In effect, mutations can be thought to be poured down the branches of the phylogenetic tree independently, to give us the cladogram (see Tavaré (2003)). Phylogenetic trees are typically used for a small number of taxa or individuals aiming to reconstruct their history.

Phylogenetic trees also contain information about the *time* of each coalescence† event and are *binary*†, because such events cannot happen simultaneously (or rather, in continuous time such an event has zero probability), and thus the possible topologies of the tree are fewer. By tree topology here we mean the sequence according to which nodes coalesce (i.e. have their most recent common ancestor), but not information about the length of time these took. A lot of research has been done lately to find efficient methods of reconstruction of phylogenetic trees. The main programs written to do this are MrBayes, BAMBE and RAxML. Below we present a few approaches.

Method described by Mau et al. (1999)

The aim is to model the stochastic process of evolution resulting to, say s species (i.e. our data). A phylogenetic tree records the path of evolution from a single ancestral population to the present array of n populations under study. There are two sub-processes to look at: the first one is, starting at the root of the tree, is the occurrence of mutations. The second one is the branching, which is when one of the mutations which occurred creates a branch leading towards only one taxon† (or group of taxa). The node at which the branching happens represents the Most Recent Common Ancestor (MRCA) of the two groups of taxa formed by the two branches. In this way the phylogeny is characterised by $\tau = (t, \sigma)$, where $t = (t_1, t_2, ..., t_{n-1}) \in \mathbb{R}^{n-1}$ is the sequence of speciation times, and σ is a permutation of $\{1, 2, ..., n\}$

Before we describe the MCMC algorithm used, we discuss the assumptions of the model and its parameters:

(a) First we concentrate on the stochastic process of mutations. In order to analyse DNA sequence data, one of the assumptions that we make is that the individual sites (i.e. each of the letters) are in linkage equilibrium between them, which means that they mutate independently of each other. Over time, the stochastic process of mutations may be modelled as a Poisson Process with independent evolution among branches. The method in this paper assumes a *molecular clock*†.

The model used here is the HKY85 model (Hasegawa-Kishino-Yano, see Hasegawa et al. (1985)), where unequal base frequencies are allowed and the difference between transitions and transversions is accounted for through one parameter (κ). Then the probabilities of being at state j at time t starting from state i at time 0 will be given by the matrix

$$P = e^{-Qt}$$

where Q represents the jump (or generator) matrix. The generator matrix for this process looks like:

$$Q = \theta \begin{pmatrix} \cdot & \kappa\pi_G & \pi_C & \pi_T \\ \kappa\pi_A & \cdot & \pi_C & \pi_T \\ \pi_A & \pi_G & \cdot & \kappa\pi_T \\ \pi_A & \pi_G & \kappa\pi_C & \cdot \end{pmatrix}$$

The π's indicate the long-run probabilities of each base along one very long branch and κ allows different substitution rates for transitions and transversions. These further involve an overall mutation rate θ, and all of the parameters may be different depending on codon† position (i.e. 1st, 2nd or 3rd in the codon). The parameter κ has the same role as b did in the classical approach described in the previous section.

Our stationary base probabilities can readily be estimated from the observed frequencies in our data $\hat{\pi}_G, \hat{\pi}_C, \hat{\pi}_T, \hat{\pi}_A$, as well as the overall mutation rate, by looking at the variation at each codon site.

When looking at the chain of mutations along the tree, by setting our initial base probabilities at the leaves to be equal to the equilibrium base probabilities, our chain becomes time-reversible, and so we may run the chain or calculate probabilities backwards (from leaves to root).

These assumptions mean that the calculation of the likelihood may be calculated recursively according to a *pruning* algorithm: Let u_i denote the unknown base at the site of interest in the ancestral sequence associated with internal node i at tree τ. Each internal node partitions the descendant species into two distinct groups, whose observed DNA data we label $A(i)$ and $B(i)$. Then the conditional probability of all data descending from i, given u_i is

$$p\{A(i), B(i)|u_i, \tau\} = p\{A(i)|u_i, \tau\}p\{B(i)|u_i, \tau\}$$

by independence. Here, for example, the probability that a base T is a C after time t is given by

$$\mathbb{P}(T \text{ after time } t|C \text{ at time } 0) = \left(e^{-Qt}\right)_{34}$$

If Q is diagonalisable, then we may find a set of basis vector U so that

$$Q = \mathrm{U}^{-1}D\mathrm{U}$$

where D is a diagonal matrix with elements $\lambda_1, \lambda_2, \lambda_3, \lambda_4$, the eigenvalues of the Q. Then e^{Qt} is easily computed since:

$$\begin{aligned} e^{-Qt} &= I + (-Qt) + (-Qt)^2/2! + (-Qt)^3/3! + ... \\ &= I - (\mathrm{U}^{-1}D\mathrm{U}t) + (\mathrm{U}^{-1}D\mathrm{U}\mathrm{U}^{-1}D\mathrm{U})t^2/2! - (\mathrm{U}^{-1}D\mathrm{U}\mathrm{U}^{-1}D\mathrm{U}\mathrm{U}^{-1}D\mathrm{U})t^3/3! + ... \quad (2.6) \\ &= I - (\mathrm{U}^{-1}D\mathrm{U})t + (\mathrm{U}^{-1}D^2\mathrm{U})t^2/2! - (\mathrm{U}^{-1}D^3\mathrm{U})t^3/3! + .. \\ &= \mathrm{U}^{-1}e^{-Dt}\mathrm{U} \\ &= a + b\lambda_1^t + c\lambda_2^t + d\lambda_3^t \end{aligned}$$

This means that the probabilities will be a linear combination of the powers of the eigenvalues, with the coefficients being easily computable using boundary conditions.

So, we may start at the leaves and conditioning at every step calculate the total probability from a given site which is given by:

$$\sum_{u_{\text{root}}} p\{A(\text{root}), B(\text{root})|u_{\text{root}}, \tau\}p(u_{\text{root}}).$$

Since we assume linkage equilibrium, we calculate the likelihood for the phylogeny of each single nucleotide separately and multiply them all together.

(b) Secondly we consider the speciation process, which represents the tree topology. The phylogeny does not uniquely determine the tree, as at each binary node the orientation is irrelevant (i.e. which nodes appears on left and right), so we get 2^{n-1} equivalent trees. Given a collection of inter-mutation times, $t_1, ..., t_{n-1}$ we may arrange them in $(n-1)!$ ways, and we may assign nodes in $n!$ ways. Hence we get $n!(n-1)!/2^{n-1}$ distinct labelled histories, given the inter-mutation times.

We use the following representation of tree topologies, using nested parentheses, such as

$$top(\tau) = (((1, (4, 7)), (2, (3, 4))), 5)$$

to represent the tree in Figure 2.3.

To account for the 2^{s-1} equivalence of tree topologies, the convention when joining up two groups is to place on the left the branch which contains the smallest number.

We have described the setup of our model and how to calculate the likelihood of a specific phylogeny, so now we present the MCMC steps.

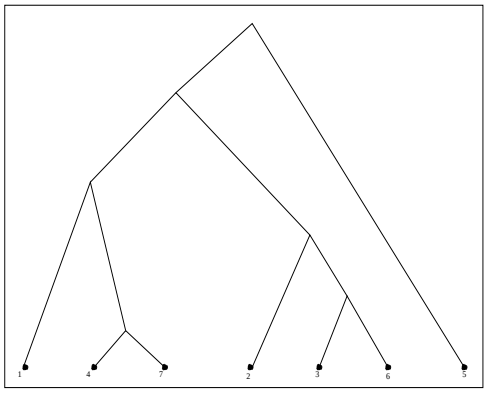


Figure 2.3: The tree topology represented by the above form

First we estimate the base frequencies and mutation and substitution parameters. Then we start from an initial tree topology (t, σ). This step obviously requires some tuning: Although we know that on average mutations happen at a constant rate, we have little evidence to suggest how many mutations occurred between any two speciation events.

We choose at random one of the 2^{n-1} equivalent trees, and perturb the inter-mutation times slightly according to a uniform random variable. In particular, starting from a vector of times t, we generate a new vector t' of times by

$$t'_i = t_i \oplus \epsilon_i, \qquad \text{for } i = 1, 2, 3, ..., n-1$$

where ϵ_i are independent identically distributed Uniform$(-\delta, \delta)$ random variables for some tuning parameter δ, and $\oplus$ indicates addition reflected into the interval $(0, t_{\max})$. Although this changes the times only by a small amount, the topology of the new tree can be very different, as changing the times means changing the branching structure.

Then we calculate the resulting posterior likelihood by using the pruning algorithm described above, and calculate the Hastings ratio

$$\begin{aligned} A &= \min\left(1, \frac{f(\text{data}|\tau)p(\tau)q(\tau,\tau')}{f(\text{data}|\tau')p(\tau')q(\tau',\tau)}\right) \\ &= \min\left(1, \frac{f(\text{data}|\tau)}{f(\text{data}|\tau')}\right) \end{aligned}$$

since proposing new times is according to a uniform and hence is symmetric, and we use a uniform prior for the trees.

The above proposal method ensures that the tree proposed is quite 'near' the current tree, however such a proposal also allows for enough mixing as the candidate tree can have quite a different branching structure even though the likelihood will be similar.

By the above algorithm we obtain a distribution on the phylogenetic trees, which may now be used in order to carry out the nested clade analysis. We analyse each tree separately, and then get a collection of results.

Method described by Newton et al. (1999)

In this paper (by the same authors as the previous one) a very similar method is described, from a slightly different perspective. Evolution has two components that may be modelled as a stochastic process: the branching created by speciation and extinction to form a phylogeny, and the propagation of characters along the branches of that phylogeny. In this method we choose to treat the phylogeny as a parameter in a model for the propagation of data along each lineage. The model assumed for mutations is again HKY85.

We consider a weighted tree Ψ, in which each edge has an associated positive weight. The branch lengths (edge weights) are the vertical distances between connected nodes. The ordering in which the mergings occur define coalescent levels, whereas the times at which these mergings

occur denote coalescent times. Such a tree can be uniquely defined either by its topology and branch lengths or by its labelled history and coalescent times. The number of topologies and labelled histories grows rapidly with n, equal to $(2n-3) \times (2n-5)... \times 1$ (inductively) and $n! \times (n-1)!/2^{n-1}$ respectively.

We form the matrix whose entries are determined by the within-tree distances between leaf nodes. Each permutation of the leaves generates a different matrix, and a rooted tree where all leaf nodes are equidistant from the root is called *cophenetic*. Clearly, such matrices are composed of at most n distinct entries.

Choosing the orientation we chose before, i.e. upon merging two groups, we place on the left the one containing the smallest node, we define a canonical ordering. A cophenetic matrix with a canonical ordering has the important property that its super-diagonal (the diagonal of the sub matrix formed when deleting the first column and nth row) contains each distinct non-zero cophenetic distance. Below is an example of a canonical cophenetic matrix, describing the distances for the tree in Fig. 2.4, where coalescent times T are set at (0.8,0.3,0.7,0.5,0.9,1.5):

	5	7	4	1	2	6	3
5	0	9.4	9.4	9.4	9.4	9.4	9.4
7		0	1.6	4.6	6.4	6.4	6.4
4			0	4.6	6.4	6.4	6.4
1				0	6.4	6.4	6.4
2					0	3.6	3.6
6						0	2.2
3							0

Here notice that $2.2 = 2 \times (t_1 + t_2)$ so that the distance between 3 and 6 is the vertical distance that has to be travelled to get from 3 to 6 on the cladogram (so up and down again). 5 is the last one to coalesce and so the distance from all other nodes is equal to 9.4, which is twice the total height of the tree.

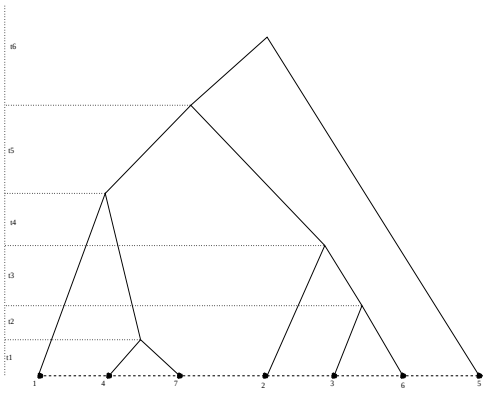


Figure 2.4: Sample phylogeny on seven taxa

A stochastic model describes the joint distribution of $y_v,\ v \in V = \mathcal{I} \cup \mathcal{L}$, the historical record at $\mathcal{I}$ and current status at $\mathcal{L}$ for a given site. This method describes a different way to represent phylogenies by considering the stochastic process above, and thus a different way of proposing and updating trees. More detail may be found in Mau et al. (1999).

Method described by Larget and Simon (1999)

The point of this paper is that it suggests a new approach for updating the trees, which is local rather than global, and also it extends the global methods described above to the case when we don't assume a molecular clock.

(a) GLOBAL method with a molecular clock: Equivalent to the method described above

(b) GLOBAL method without a molecular clock: We perturb the $2(n-1)$ branch distances instead of $n-1$. This is because at a coalescence event, the two branches joining up do not need to have the same length. Since the likelihood methods we use are reversible and do not distinguish between alternative rootings of the tree, instead of changing the tree we just pick an alternative rooting.

(c) LOCAL method with a molecular clock: In this case when proposing a new tree, we pick at random an internal edge from our tree (i.e. not joined to any of the leaves) and we rearrange the nodes which it joins up at random according to some algorithm based on the lengths of the edges replaced.

(d) LOCAL method without a molecular clock: A similar procedure is followed in the case of no molecular clock on the unrooted tree.

Method described by Altekar et al. (2004)

In this paper a Metropolis-Coupled Markov Chain Monte Carlo ($(MC)^3$) method is suggested which deals with the problem of slow mixing and of a standard MC chain getting stuck in local optima. The method is similar to simulated tempering. Two chains run in parallel, a "hot" and a "cold" one, and the overall chain jumps between the two.

Example: Simulated dataset

We use MrBayes (the latest package for estimating phylogenies, following the method described by Altekar et al. (2004)) to estimate the phylogenetic tree for the simulated dataset. We use the graphics software package Mesquite to obtain a graphical representation of the results are given below. The numbers on the branches are the estimated probabilities according to which that branch is correct. Using the sequences, the estimated phylogeny can be transformed to give us the true cladogram of Figure 1.3.

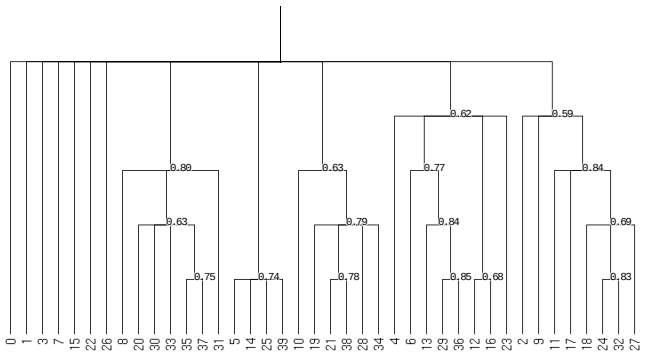


Figure 2.5: Output of MrBayes for the simulated dataset

Example: The beetle data

We use MrBayes to estimate the phylogenetic tree for the Beetle data. The results are given below, again using Mesquite, in Figure 2.6. In this case it is not quite as straightforward to infer the cladogram from the phylogenetic tree.

2.3 Forming the Nested Cladogram

Before we carry out the association study, we need to form the nested cladogram. Once we have a cladogram, and we want from it to obtain the nested one. The nesting algorithm is as follows (Templeton et al. (1987)): We start with our 0-step clades, which are just the haplotypes represented as leaves in our cladogram. Given the n-step clades, the $n+1$-step clades are formed

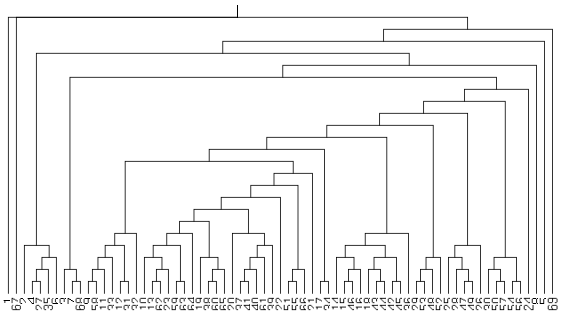


Figure 2.6: Output of MrBayes for the beetle dataset

by taking the union of all n-step clades which can be joined up by moving one mutational step back from the terminal node of each n-step clade. We continue the process recursively until all the nodes in our cladogram have been nested. The nesting process is easier understood through an example:

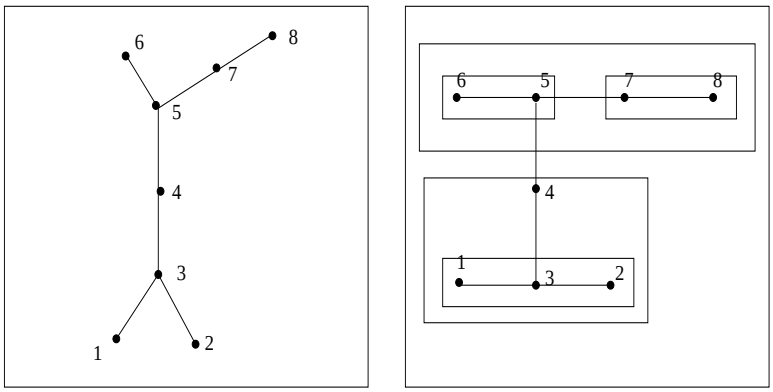


Figure 2.7: An example of a cladogram, unnested (left) and nested (right).

In the above example we proceeded as follows: First we joined up leaves to their neighbours. In the case where two leaves are joined to the same neighbour, the two leaves are nested together (i.e. clade 1-3-2). So we obtain 5-6, 7-8 and 1-3-2 as our 1-step clades. In effect, these 3 groups are collapsed onto only 1 node each, behaving like a leaf: we now have 5,7 and 3 instead of the whole 3 groups(7, 8, 1, 2 were 'chopped'). In the next step, we join up 7 (and the nodes associated with it from the previous step) and 5 (likewise), and we also join up 3 and 4 (as shown in diagram). Finally, our last step only involves joining the whole thing together.

The procedure described above is not always well-defined, as there are special cases which are ambiguous, or where nesting is not complete. For example, in the figure below, on the first step we nest 1-2 and 4-5, leaving 3 stranded (as the next nesting step is just nesting the whole thing which has no practical use in terms of the analysis).

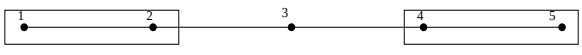


Figure 2.8: An example of incomplete nesting

In these cases, special guidance is given about how to proceed (see Templeton and Sing (1993)). If that node (or clade) represents an unobserved haplotype, then it is simply left ungrouped with no complications. However, if it belongs to our sample, it needs to be grouped with another clade, and we use the following guidelines: First, the stranded clade should be grouped with the nesting category that has the smallest sample size because such a grouping tends to maximise statistical power. Secondly, if the smallest sample size is observed in more than one alternative,

then the stranded clade should be nested with the alternative to which it is connected through a non-restriction site mutation. Non restriction site mutations tend to be unique more often than restriction site mutations, so the connection involving the non restriction site change is in general a more certain connection.

2.3.1 Example: The Beetle data from La Palma

We nest the Figure 2.1 used in the previous section to obtain the following nested cladogram. We have 4 levels of nesting, with 3 groups of nodes at the 4th level.

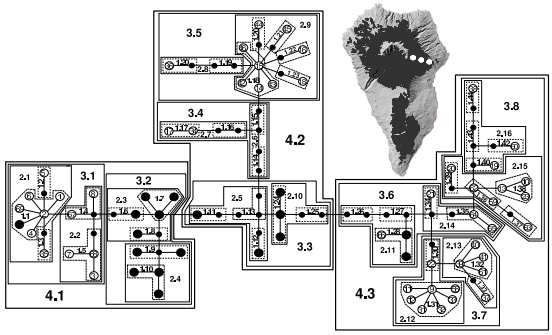


Figure 2.9: The nesting of the cladogram in Figure 2.1

2.3.2 Example: Simulated dataset

In the case of the simulated dataset, we obtain 4 levels of nesting, with 4 groups at the 4th level.

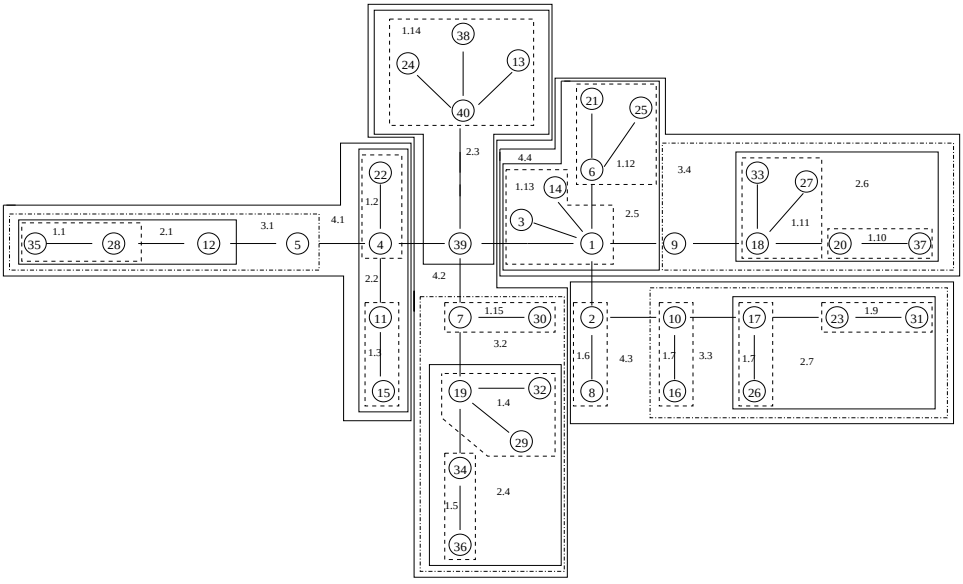


Figure 2.10: The nesting of the cladogram in Figure 1.3

2.4 Carrying out Nested Clade Analysis

Our aim is to use the nested cladogram in order to associate it with significant (or insignificant) geographical or phenotypic data. For example, if haplotypes 24, 38, 13, 40 in the simulated dataset have a certain phenotype and everyone else has a different one, then we expect to conclude that the mutation $39 \to 40$ caused this effect. The main assumption in NCA is that if an undetected mutation causing a phenotypic effect occurred at some point in the evolutionary history of the population, it would be embedded in the same historical structure represented by the cladogram. In other words, even if we don't detect some mutation, the evolutionary history we are predicting would still be correct, and so the hidden mutation would only be present in a certain clade. For example, if we have assumed that the haplotypes 39 and 40 were one mutation apart, but in reality we failed to detect a hidden mutation between them, so that in fact the process was $39 \to 0 \to 40$, then the phenotypic effect will still have the same structure and so the same clades will appear significant.

Haplotypes at the tip of the tree represent younger ones, and also specific mutations may be associated with a specific isolated geographical region, if the mutation happened after isolation. Similarly, some phenotypic effects may be associated with a specific mutation, and we expect that to appear in the nested cladogram. It is much more efficient to test a specific phenotypic effect with a mutation, by looking at adjacent haplotypes, rather than looking at the whole tree itself and compare to every single possible combination.

How to use the nested cladogram to associate with phenotypic data Starting from level 1, at each level, an ANOVA is performed, and Residual Sum of Squares (RSS) contributions are examined to identify clades which contribute most. To avoid the possibility of "overspill", where the effect of significant mutations carries through to different clades and masks what is really happening, significant clades are examined and compared using Bonferroni comparisons.

One of the main uses of the nested cladogram is associating specific mutations with some phenotypic effect. The classical way described by Templeton in Templeton et al. (1987) is to look separately at each n-step clade and use e.g. 1-step clades as our factors and carry out an ANOVA on the quantitative trait observed in these clades.

2.4.1 Example: Simulated phenotypic dataset

We use the cladogram of Fig. 1.3 and generate a phenotypic effect, selecting edges 6 and 27 (joining nodes 39-7 and 9-8) to be significant. For each node, there are $1 + \text{Poisson}(4)$ data points, i.e. observed phenotypes of individuals. The 3 subgraphs have normal distributions $N(0,1)$, $N(0.8,1)$ and $N(-0.8,1)$ and the two significant separating edges are 6 and 27, joining FIND.

Looking at the overlap of the distributions, we see that there is a large common area, making this test quite tough. In fact, the $N(0,1)$ and $N(0.8,1)$ densities have a common area of ≈ 0.69, and $N(0.8,1)$ and $N(-0.8,1)$ an area 0.42. We expect it to be difficult to detect the exact mutation which is associated with this effect.

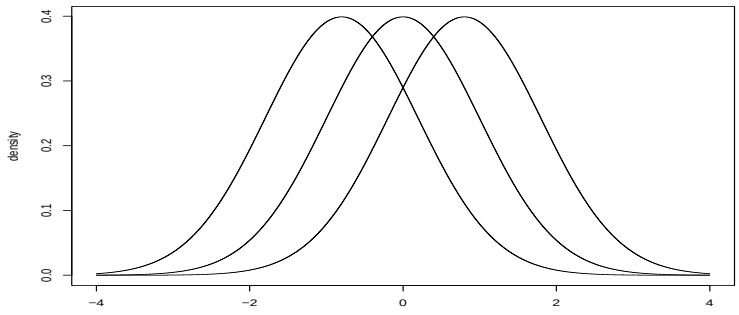


Figure 2.11: The density function of $N(0,1)$, $N(-0.8,1)$ and $N(0.8,1)$

Carrying out the NCA on the nested design, we get some significant results from the clades which are indeed significant (the details are not shown here). However, the precise mutations which are associated with the significant phenotypic effect are not clearly detected.

How to use the nested cladogram to draw geographical conclusions Once we have formed the cladogram, the problem arise of how to use it to draw geographical conclusions. There are 3 major biological factors that can cause a significant spatial/temporal association of haplotype variation: a) restricted gene flow† b) past fragmentation† and c) range expansion (including colonisation).

The method described by Templeton (1998) states one should first test for association between the clade distance and the nested clade distance, in a similar way as before. The geographical data are quantified in two ways: the clade distance D_c, which represents the geographical range of a clade, and D_n, which measures how that particular clade is geographically distributed relative to its closest evolutionary clades (i.e. clades in the same higher-level category). Specifically, D_c is the average distance of haplotypes from that clade from the geographical centre of the clade. D_n is the average distance of a haplotype form that clade to the geographical centre of all higher-level clades which contain the clade in question. Both these distances are a measure of the spatial spread of a clade, where only physically feasible paths are taken into account (i.e. routes which are possible for the species to have taken)

If the null hypothesis of no association is rejected, then there is a descriptive inference key (see Appendix in Templeton (1998)) suggested which has a very high success rate at giving the correct answer to what happened spatially and/or temporally, as shown using well-known and analysed examples.

2.4.2 Example: The Beetle data, a phylogenetic study

Emerson et al. (2005) carry out the NCA on the beetle data, using the cladogram derived above (see Figure 2.1). On the 1st level, clade 1.1 is the only one that appears significant. On level 2, clades 2.1 and 2.3 and 2.14, whereas in level 3 clades 3.2, 3.6, 3.7. In the last level, all clades are significantly different. Emerson et al. (2005) then use the descriptive inference key from Templeton (1998) to draw conclusions about the phylogeography of the beetles on La Palma.

Chapter 3

A new approach

We describe an MCMC method of analysing sequence data in which all the steps are carried out simultaneously so that the uncertainty carries through. We also try to include more factors in our model to make our results more reliable.

3.1 Estimating the rooted cladogram for the simple case

The first thing we have to do in order to estimate the cladogram is to decide which node is its root.

3.1.1 Finding the root

In the most simple case, we have a set of haplotypes forming a connected tree. Since no homoplasies or recombinations are present, the tree is deterministic. However, we have to make inference about the root r of the tree. By the term degree of a node here we mean the number of edges connected to that node. An old haplotype is more likely to have higher degree (i.e. to have mutated many times) in the graph. Since we are not making any inference about branch lengths in the cladogram (representing mutation jump times), we use the degree of each node as a measure of its age. We want to investigate the relationship between a node's degree and its probability of being the root.

In a set of n nodes, the probability of a node being the root given its degree is given by

$$\mathbb{P}(\text{root}|\text{degree}) = \frac{\mathbb{P}(\text{degree}|\text{root})\mathbb{P}(\text{root})}{\mathbb{P}(\text{degree})} = \frac{\mathbb{P}(\text{degree}|\text{root})}{n\,\mathbb{P}(\text{degree})}$$

using Bayes' rule and assuming that, given no information about the nodes, any node is equally likely to be the root.

Clearly, a node having degree r has different significance in a set of 50 or 500 nodes. We want to work out a distribution which will be independent of the number of nodes in the graph. For that reason, we look at the "rank" of each node's degree. That is, if a node has the highest degree, it has rank 1, if 2nd highest then rank 2 and so on.

We expect an exponential relationship between $\mathbb{P}(\text{root}|\text{rank})$ and the rank, so we look at:

$$\log \mathbb{P}(\text{root}|\text{rank}) = \log\left(\frac{\mathbb{P}(\text{rank}|\text{root})}{n\,\mathbb{P}(\text{rank})}\right)$$

We simulate 100,000 trees to estimate these probabilities for $\alpha = \zeta = 2$, $\beta = \gamma = \delta = \epsilon = 1$ (see Sequeira et al. (2000); Jordal et al. (2000); Scataglini et al. (2005)) and $\pi_i = 0.25, 0.25, 0.25, 0.25$, and get the following results (see Figure 3.1):

Carrying out a few more simulations we notice that whatever the values of $\alpha, \ldots, \zeta$ and π_i's are, the relationship between a node's probability of being the root and its degree's rank is the same. That is to be expected, since whatever the values of the π_i's and $\alpha, \ldots, \zeta$ are, the π_i's define a stationary distribution for the Q-matrix. So, whatever the initial sequence (drawn from the equilibrium distribution), the next sequence will have the same distribution as the root.

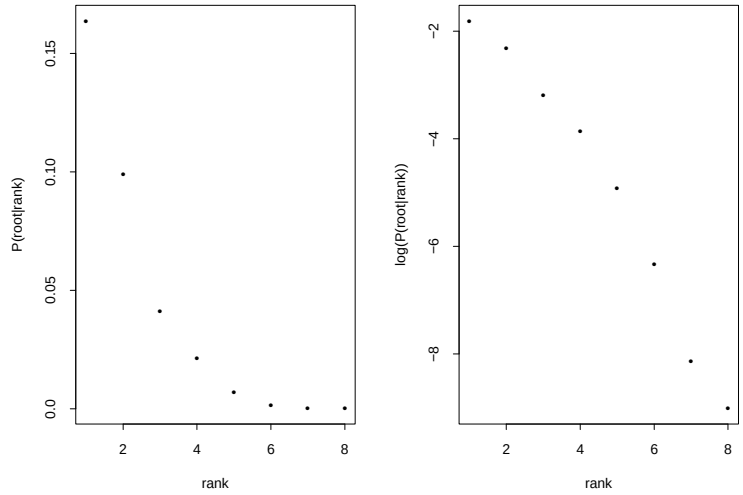


Figure 3.1: Simulated conditional distribution

This is because the process can be viewed as follows: We start off with one sequence, whose amino-acids are at the stationary distribution. Then, since all the sites have the same distribution, they all have the same probability of being the first to mutate, namely $1/l$ (where l is the length of the sequence). Once one of the sites has mutated, we now have two sequences present. By the Strong Markov property, the first process (corresponding to the initial sequence) starts afresh. Moreover, the new sequence is also at the stationary distribution. Hence all sites on both sequences have the same probability of being the first to mutate, namely $1/2l$. Clearly, this process continues in the same manner, and the specific values of the Q-matrix do not affect these probabilities, as long as we start off at the stationary distribution. In fact it can be easily shown that

$$\mathbb{P}(\text{root degree=r}) = \frac{1}{n} \sum_{\substack{i_1,\ldots,i_{r-1} \\ \text{distinct}}} \frac{1}{i_1}\frac{1}{i_2}\cdots\frac{1}{i_{r-1}}$$

independent of the transition matrix, provided the chain is at equilibrium.

Clearly, more than one node may have the same degree and hence the same degree rank.

Step A1: Propose a new root r' at random $\propto$ order (a different proposal would be to propose a new root r' at random from the nodes adjacent to the current root) and accept with probability

$$A_r = \frac{L(D|r')p(r'|\text{rank})}{L(D|r)p(r|\text{rank})}\frac{\text{degree}(r)}{\text{degree}(r')} \tag{3.1}$$

In fact, if the move $r \to r'$ is along an edge $A \to G$ then A_r becomes:

$$A_r = \frac{\mathbb{P}(G \to A)p(r'|\text{rank})}{\mathbb{P}(A \to G)p(r|\text{rank})}\frac{\text{degree}(r)}{\text{degree}(r')} \tag{3.2}$$

since the direction of all other edges remains the same.

A proposal which only allows moves to adjacent nodes has the advantage that the nodes proposed are more likely to be roots than a randomly selected one, since in general adjacent nodes are of similar age and hence of similar probability of being the root. The disadvantage of this that the chain gets easily stuck in certain parts of the graph, so a more efficient proposal in some cases is to propose a root randomly from our graph with probability $\propto$ to the degree of each node. It is easy to check that the acceptance probability stays the same as above. To make the algorithm more efficient, we order the nodes so that their degrees are descending before the start of the chain.

3.1.2 Updating the amino acid frequencies

Having picked a root, our tree becomes directed and obtaining a measure of its likelihood is easier. We describe an algorithm to update the gene frequencies.

We have parameters $B_1, \ldots, B_4$ such that $\mathbb{E}B_i = 1 \quad \forall i$

In fact we take

$$B_i \sim \text{Normal}\left(1, \sigma_B^2\right) \quad \forall i$$

We use a Dirichlet prior on our probabilities $(\pi_1, \ldots, \pi_4)$ since it is the conjugate prior of the multinomial distribution:

$$(\pi_1, \pi_2, \pi_3, \pi_4) \sim \text{Dirichlet}(B_1, B_2, B_3, B_4)$$

Here the parameters B_i represent our prior belief about the relative relationship of amino-acid frequencies. If the prior is rather uncertain then we can use a Jeffrey's prior so that $B_i = 0.25$ for $i = 1, 2, 3, 4$, and the data which typically contains a large number of amino-acid values will strongly dominate. In these cases, the parameters B_i are not updated but held constant at 0.5.

Then, given the data and a tree topology T, the distribution becomes:

$$\begin{aligned} &(\pi_1, \pi_2, \pi_3, \pi_4)|D, \boldsymbol{B} \\ &\sim \text{Dir}(B_1 + n_1, B_2 + n_2, B_3 + n_3, B_4 + n_4) \end{aligned} \tag{3.3}$$

Here the n_i's represent the number of observed genes of type i within the data D.

Step A2: So, first we want to propose the parameters B_i using some proposal distribution. They do not need to sum to one, so we can update them individually. This is most easily achieved by updating each of the B_i in turn by generating $d \sim N(0, \sigma_d^2)$ and setting $\log B_i' = \log B_i + d$. This move is then accepted with probability $\alpha(B_i, B_i') = \min(1, A_2)$ where

$$A_2 = \frac{p(\boldsymbol{\pi}|\boldsymbol{B'})p(\boldsymbol{B'})}{p(\boldsymbol{\pi}|\boldsymbol{B})p(\boldsymbol{B})}$$

Then, substituting, A_2 becomes:

$$\begin{aligned} A_2 = \prod_{i=1}^{4} \quad \Bigg\{ \quad & \exp\left(-\frac{1}{2\sigma_m^2}\left[(\log B_i' - \mu_{B_i'})^2 - (\log B_i - \mu_{B_i})^2\right]\right) \\ \times \quad & \pi_i^{B_i' - B_i} \frac{\Gamma(\sum B_i - B_i + B_i')\Gamma(B_i)}{\Gamma(\sum B_i)\Gamma(B_i')} \Bigg\} \end{aligned}$$

where $\mu_{B_i} = 1 \quad \forall i$.

Once we have our new value for $\boldsymbol{B}$ we continue with proposing $\boldsymbol{\pi}$ and a tree topology T:

Step A3: Given $\boldsymbol{B}$ and the data, we update $\boldsymbol{\pi}$ by generating from the posterior

$$\begin{aligned} &(\pi_1, \pi_2, \pi_3, \pi_4)|D, \boldsymbol{B} \\ &\sim \text{Dir}(B_1 + n_1, B_2 + n_2, B_3 + n_3, B_4 + n_4) \end{aligned} \tag{3.4}$$

3.1.3 Updating the transition/transversion probabilities

In order to calculate the likelihood of a tree, we need to estimate the jump probabilities, which depend upon the relative mutation rates. For example,

$$\mathbb{P}(A \to G|A) = \kappa\pi_G\alpha = \frac{\pi_G\alpha}{\pi_G\alpha + \pi_C\beta + \pi_T\gamma}$$

and so, conditional on the tree topology (and hence the total number of nodes), and the root, the likelihood of the tree will be of the form

$$\propto \prod_{\substack{i,j=A,G,T,C \\ i \neq j}} (\mathbb{P}(i \to j)|i)^{n_{i \to j}}$$

If the total number of nodes is not fixed, then a problem arises during the calculation of the likelihood of the tree topology. The normalisation constant to account for different sized trees is very hard to calculate, and so instead of looking at the probability of a transition set $\boldsymbol{S}$ given a tree topology T, we assign a prior to the different tree topologies and look at $\mathbb{P}(\boldsymbol{S}, T) = \mathbb{P}(\boldsymbol{S}|T)\mathbb{P}(T)$. To begin with we can take a uniform prior over all tree topologies of the same size and a uniform prior over possible sizes of trees. In the examples considered here we only consider trees of equal size, and hence do not deal with this situation.

Depending on our assumptions, we discuss 2 cases:

1. Assuming equal mutation rates across amino acids and equal gene frequencies the Q-matrix can be written as
$$Q = \theta \begin{pmatrix} -1 & \alpha & \beta & \gamma \\ \alpha & -1 & \delta & \epsilon \\ \beta & \delta & -1 & \zeta \\ \gamma & \epsilon & \zeta & -1 \end{pmatrix}$$
Here the parameter θ represents an overall mutation rate so that the mutation rates across rows add up to 1 rather than some other constant. We assume priors
$$\alpha,\ \zeta \sim \mathrm{N}(0.5, \sigma_s^2),$$
$$\beta,\ \gamma,\ \delta,\ \epsilon \sim \mathrm{N}(0.25, \sigma_s^2),$$
and that transition/transversion rates are independent. Here the transition/transversion ratio is on average 2, appropriate for mitochondrial DNA of beetles, and we should adjust appropriately if using a different locus or species.
Then, starting with $(\alpha, \ldots, \zeta)$ so that
$$\begin{aligned} \alpha + \beta + \gamma &= 1 \\ \alpha + \delta + \epsilon &= 1 \\ \beta + \delta + \zeta &= 1 \\ \gamma + \epsilon + \zeta &= 1 \end{aligned}$$
we can propose new parameters by setting
$$\begin{pmatrix} \alpha' \\ \beta' \\ \gamma' \\ \delta' \\ \epsilon' \\ \zeta' \end{pmatrix} = \begin{pmatrix} \alpha + \epsilon_1 \\ \beta + \epsilon_2 \\ \gamma - \epsilon_1 - \epsilon_2 \\ \delta - \epsilon_1 - \epsilon_2 \\ \epsilon + \epsilon_2 \\ \zeta + \epsilon_1 \end{pmatrix}$$
so that
$$\begin{aligned} \alpha' + \beta' + \gamma' &= 1 \\ \alpha' + \delta' + \epsilon' &= 1 \\ \beta' + \delta' + \zeta' &= 1 \\ \gamma' + \epsilon' + \zeta' &= 1 \end{aligned}$$
With probability $1/2$ we set $\epsilon_2 = 0$ and
$$\epsilon_1 \sim U[\max(-\alpha, -\zeta, \gamma - 1, \delta - 1, -E_1), \min(1 - \alpha, 1 - \zeta, \gamma, \delta, E_1)]$$
where E_1 is a constant, and with probability $1/2$ we set $\epsilon_1 = 0$ and
$$\epsilon_2 \sim U[\max(-\beta, -\epsilon, \gamma - 1, \delta - 1, -E_1), \min(1 - \beta, 1 - \epsilon, \gamma, \delta, E_1)]$$

Step A4: We accept or reject the new values according to the probability of this move which is $\alpha(\boldsymbol{\alpha}, \boldsymbol{\alpha'}) = \min(1, A_4)$ where

$$A_4 = \frac{\mathbb{P}(D, T, r|\boldsymbol{\alpha'})}{\mathbb{P}(D, T, r|\boldsymbol{\alpha})} \frac{p(\boldsymbol{\alpha'})}{p(\boldsymbol{\alpha})} \frac{q(\boldsymbol{\alpha'} \to \boldsymbol{\alpha})}{q(\boldsymbol{\alpha} \to \boldsymbol{\alpha'})}$$

2. Assuming free mutation rates and amino acid frequencies: Assume priors so that

$$\alpha,\ \zeta \sim \mathrm{N}(2, \sigma_m^2)$$

$$\beta,\ \gamma,\ \delta,\ \epsilon \sim \mathrm{N}(1, \sigma_m^2).$$

Again, here the mean transition/transversion ratio is specific to beetles and their mitochondrial DNA.

Update one of $\alpha, \ldots, \zeta$ randomly by a proposal $\alpha \to \alpha + \epsilon$ where $\epsilon \sim U[\max(-E_2, -\alpha), E_2]$. so that the probability of the move is again

$$A_4 = \frac{\mathbb{P}(D, T, r|\boldsymbol{\alpha'})}{\mathbb{P}(D, T, r|\boldsymbol{\alpha})} \frac{p(\boldsymbol{\alpha'})}{p(\boldsymbol{\alpha})} \frac{q(\boldsymbol{\alpha'} \to \boldsymbol{\alpha})}{q(\boldsymbol{\alpha} \to \boldsymbol{\alpha'})}$$

so we carry out step A4 above.

So, the algorithm in the simple case where the tree topology is deterministic follows the steps A1-A4.

3.2 Phenotypic and Phylogeographic analyses

3.2.1 Phenotypic Analyses for 1-dimensional traits in the simple case

We want to partition phenotypic data but only allow clusters consistent with the removal of K edges from the graph.

First we assume that our data forms a connected graph and involves no homoplasy or recombination. This means that the number of distinct haplotypes is smaller than the number of variable restriction sites and that the underlying tree is uniquely defined. Each edge has a number associated with it, namely the restriction site on which the mutation happened, and all the edges have distinct numbers associated with them.

We have a sample of size n from some real-valued phenotypic effect from the haplotype structure. We are trying to find the underlying significant mutation(s), and we know that there are K causal mutations. We also assume that our data is Normally distributed, with $K+1$ different means $\mu_{1true},\ \mu_{2true}, \ldots, \mu_{(K+1)true}$ depending on presence or not of the causal mutations, and that we have a common underlying variance σ^2_{true} . Then if these mutations are represented by edges $\boldsymbol{e}_{true}$, the resulting $K+1$ clusters would be significantly different.

We introduce the following notation: We start off with data D with a total of n data points in $\mathbb{R}$, each data point corresponding to one of $m \leq n$ haplotypes (i.e. nodes on a tree), and we denote by $\boldsymbol{e}$ the proposed set of edges representing the causal mutations. Here $\overline{x}_i$ denotes the sample mean of cluster i and n_i the sample size of cluster i, so that $\sum n_i = n$, the total sample size. Similarly, $\boldsymbol{\mu}$ denotes the proposed vector of underlying cluster means, and σ^2 the proposed underlying variance. Lastly, we denote the cluster to which node i belongs by c_i.

We assume the following distributions:

$$\begin{aligned}
\tau = \sigma^{-2} &\sim \mathrm{Gamma}(a, b) \\
\mu_i &\sim \mathrm{N}(0, \sigma_b^2) \\
e_i &\sim \mathrm{U}\{1, \ldots, n-1\} \text{ without replacement} \\
D|\boldsymbol{e}, \sigma, \boldsymbol{\mu} &\sim \mathrm{N}(\mu_i, \sigma^2)
\end{aligned}$$

This gives that

$$\begin{aligned}\tau|\boldsymbol{e}, D, \boldsymbol{\mu} &\sim \text{Gamma}\left(a + \frac{n}{2}, \frac{\sum (x_i - \mu_{c_i})^2}{2} + b\right) \\ \mu_i|D, \boldsymbol{e}, \sigma &\sim \text{N}\left(\frac{\overline{x_i}\,\tau n_i}{\tau n_i + \frac{1}{\sigma_b^2}}, \left(\tau n_i + \frac{1}{\sigma_b^2}\right)^{-1}\right)\end{aligned}$$

We assume the number of mutations which are causal is, say, K. We will run our MCMC chain for a few different K's later on. The chain works as follows:

We start off with an underlying variance $1/\tau^{(t)}$, causal edges $\boldsymbol{e}^{(t)}$ which gives us a disconnected tree by separating our connected graph into $K+1$ bits, of size $n_i^{(t)}$, and with $K+1$ different means $\mu_i^{(t)}$.

Step B1a: First we update $\boldsymbol{e}$. We pick one of the K edges e_i randomly and change it to another one randomly from our graph with a uniform proposal (not allowing edges which are already in $\boldsymbol{e}$). To ensure that the splitting of the clusters is done identically every time we delete the same set of edges, we order the deleted edges so that there is only one possible ordering every time. Hence we have $\binom{n-1}{K}$ possible states to jump to, rather than $K!\binom{n-1}{K}$ as we would have if two orderings were allowed.

A more efficient proposal for $\boldsymbol{e}$ is to propose an edge uniformly from the edges adjacent to the current ones. The disadvantage of this proposal is that the chain moves slowly to and from distant regions in the graph.

Step B1b: We then separate the $K+1$ disconnected graphs which result from this, and calculate their means $\overline{x}_i$ and their sizes n_i.

Step B1c: We pick τ' from a distribution

$$\tau|D, \boldsymbol{e}, \mu_i = \overline{x}_i \;\forall\; i \sim \text{Gamma}\left(a + \frac{n}{2}, \frac{\sum (x_i - \overline{x}_{c_i})^2}{2} + b\right)$$

This is an approximation to the posterior conditional $\tau|D, \boldsymbol{e}, \boldsymbol{\mu}$ with $\mu_i = \overline{x}_i$.

Here a and b are taken small so that we have a large variance, say $a = b = 0.001$.

Step B1d: We then generate μ_i' from $\mu_i|D, \boldsymbol{e}, \tau$ above. So here we have a Metropolis-Hastings step with an independent sampler for $\boldsymbol{e}$, an approximate Gibbs sampler for τ and a Gibbs sampler for the μ_i's.

Step B1e: We need to accept or reject $\boldsymbol{e},\ \tau,\ \boldsymbol{\mu}$. We have

$$\pi(\boldsymbol{e}', \tau', \boldsymbol{\mu}'|D) \propto f(D|e', \tau', \boldsymbol{\mu}')p(\boldsymbol{\mu}')p(\tau')p(\boldsymbol{e}').$$

Therefore our acceptance ratio is given by

$$\begin{aligned}A &= \frac{\pi(\boldsymbol{e}', \tau', \boldsymbol{\mu}'|D)q(\boldsymbol{\mu}' \to \boldsymbol{\mu})q(\tau' \to \tau)q(\boldsymbol{e}' \to \boldsymbol{e})}{\pi(\boldsymbol{e}, \tau, \boldsymbol{\mu}|D)q(\boldsymbol{\mu} \to \boldsymbol{\mu}')q(\tau \to \tau')q(\boldsymbol{e} \to \boldsymbol{e}')} \\ &= \frac{f(D|\boldsymbol{e}', \tau', \boldsymbol{\mu}')}{f(D|\boldsymbol{e}, \tau, \boldsymbol{\mu})}\frac{p(\boldsymbol{e}')}{p(\boldsymbol{e})}\frac{p(\tau')}{p(\tau)}\frac{p(\boldsymbol{\mu}')}{p(\boldsymbol{\mu})}\frac{\pi(\boldsymbol{\mu}|\tau, \boldsymbol{e})}{\pi(\boldsymbol{\mu}'|\tau', \boldsymbol{e}')}\frac{\pi(\tau|D, \boldsymbol{e}, \overline{x}_1, \overline{x}_2)}{\pi(\tau'|D, \boldsymbol{e}', \overline{x}_1', \overline{x}_2')}\frac{p(\boldsymbol{e})}{p(\boldsymbol{e}')} \\ &= \frac{f(D|\boldsymbol{e}', \tau', \boldsymbol{\mu}')}{f(D|\boldsymbol{e}, \tau, \boldsymbol{\mu})}\frac{p(\tau')}{p(\tau)}\frac{p(\boldsymbol{\mu}')}{p(\boldsymbol{\mu})}\frac{\pi(\boldsymbol{\mu}|\tau, \boldsymbol{e})}{\pi(\boldsymbol{\mu}'|\tau', \boldsymbol{e}')}\frac{\pi(\tau|D, \boldsymbol{e}, \overline{x}_1, \overline{x}_2)}{\pi(\tau'|D, \boldsymbol{e}', \overline{x}_1', \overline{x}_2')}\end{aligned}$$

Here

$$
\begin{aligned}
f(D|\mathbf{e}', \tau', \boldsymbol{\mu}') &= \prod_{i=1}^{n} \sqrt{\frac{\tau'}{2\pi}} \exp\left(-\frac{\tau'(x_i - \mu'_{c_i})^2}{2}\right), \\
p(\boldsymbol{e}') &= \prod_{i=1}^{K} \frac{1}{n-i+1}, \\
p(\tau') &= \frac{\tau'^{a-1} b^a \exp(-b\tau')}{\Gamma(a)}, \\
p(\mu_i) &= \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left(-\frac{\mu_i^2}{2\sigma_b^2}\right),
\end{aligned}
$$

and it can easily be calculated that

$$
\begin{aligned}
\pi(\boldsymbol{\mu}'|\boldsymbol{e}', \tau') &= \prod_{i=1}^{K+1} \sqrt{\frac{n_i\tau' + \frac{1}{\sigma_b^2}}{2\pi}} \exp\left(-\frac{1}{2}\left(\mu'_i - \frac{\tau' n_i \overline{x}'_i}{\tau' n'_i + 1/\sigma_b^2}\right)^2 \left(\tau' n'_i + \frac{1}{\sigma_b^2}\right)\right), \\
\pi(\tau'|D, \boldsymbol{e}', \overline{\boldsymbol{x}}'_1) &= \frac{\tau^{a+\frac{n}{2}-1} \left(\frac{\sum (x_i - \overline{x}_{c_i})^2}{2} + b\right)^{a+\frac{n}{2}}}{\Gamma(a+\frac{n}{2})} \exp\left\{-\left(\frac{\sum (x_i - \overline{x}_{c_i})^2}{2} + b\right)\tau\right\}.
\end{aligned}
$$

So, we calculate the ratio A and accept the step with probability

$$\min(1, A).$$

If we accept, we set

$$
\begin{aligned}
\boldsymbol{e}^{(t+1)} &= \boldsymbol{e}' \\
\tau'' &= \tau' \\
\boldsymbol{\mu}'' &= \boldsymbol{\mu}'
\end{aligned}
$$

otherwise we set

$$
\begin{aligned}
\boldsymbol{e}^{(t+1)} &= \boldsymbol{e}^{(t)} \\
\tau'' &= \tau_t \\
\boldsymbol{\mu}'' &= \boldsymbol{\mu}^{(t)}
\end{aligned}
$$

Step B2: We then generate

$$\tau^{(t+1)}$$

from the posterior conditional

$$\tau|D, \boldsymbol{e}^{(t+1)}, \boldsymbol{\mu}''$$

(and accept with probability 1 as in standard Gibbs sampler).

Step B3: Lastly, same as step 2, we generate $\boldsymbol{\mu}^{(t+1)}$ from

$$\mu_i|D, \boldsymbol{e}^{(t+1)}, \tau^{(t+1)}.$$

We now go back to step 1.

The above chain is clearly aperiodic since there is a positive probability of staying at the same state, and it is also irreducible since none of the probabilities are zero and so we can get from any state to any other. Hence the stationary distribution of the chain will be the desired posterior of the parameters of interest.

3.2.2 Example: Simulated 1-D dataset

Again, we use the generated set of 1-D phenotypic data corresponding to Figure 1.3 described in the previous chapter.

We run 10,000,000 iterations (with the first 500,000 as burn-in) using $a = b = 0.0001$ and $\sigma_b^2 = 100$. Indeed, looking at pairs of edges, the pair true pairs (6,27) appears highly significant:

pair of edges	posterior prob	
(6,27)	0.9778	
node	degree	root posterior prob
1	6	0.2335
4	4	0.0862
18	4	0.0853
19	4	0.0857
39	4	0.0858
40	4	0.0853
acceptance ratio 0.128310		

We also observe that the node which is the true root has indeed the highest posterior probability of being the root.

We then simulate 20 datasets as above. For each dataset, we conclude that the pair of edges with the highest posterior probability is the true significant one if the Bayes factor with the 2nd highest pair is > 3. If such a pair does not exist, our test is inconclusive. Running this for 20 datasets, we get 0 failures, 12 successes and 8 inconclusive results.

3.2.3 Phenotypic analyses for d-dimensional traits and phylogeographic analyses in the simple case

For d-dimensional data, we repeat the above procedure replacing by appropriate multivariate distributions. In the case of phylogeographic data, we simply assume a bivariate normal distribution of our observations, and proceed exactly like we would for a 2-dimensional phenotypic effect in this case.

We assume the following distributions:

$$\begin{aligned}
\Sigma &\sim \text{InvWishart}(m, \Psi) \\
\boldsymbol{\mu}_i|\Sigma &\sim \text{MVN}\left(\mathbf{0}, \frac{1}{\tau_{prior}}\Sigma\right) \\
D|\Sigma, \boldsymbol{\mu}, \boldsymbol{e} &\sim \text{MVN}(\boldsymbol{\mu}_i, \Sigma) \\
e_i &\sim \text{U}\{1, \ldots, n-1\} \text{ without replacement.}
\end{aligned}$$

which gives that

$$\begin{aligned}
\boldsymbol{\mu}_i|\Sigma, D &\sim \text{MVN}\left(\frac{n_i\overline{\boldsymbol{x}}_i}{n_i + \tau_{prior}}, \frac{1}{n_i + \tau_{prior}}\Sigma\right) \\
\Sigma|D, \boldsymbol{e}, \boldsymbol{\mu} &\sim \text{InvWishart}\Big(n + m, \Psi + \sum_{i=1}^{n} \boldsymbol{x}_i\boldsymbol{x}_i^T \\
&\qquad - \sum_{i=1}^{K+1} \left(n_i\overline{\boldsymbol{x}}_i\overline{\boldsymbol{x}}_i^T + \frac{n_i\tau_{prior}}{n_i + \tau_{prior}}(\overline{\boldsymbol{x}}_i - \boldsymbol{\mu}_i)(\overline{\boldsymbol{x}}_i - \boldsymbol{\mu}_i)^T\right)\Big)
\end{aligned}$$

Then steps B1-B3 become:

Step B′1a: First we update $\boldsymbol{e}$. We pick one of the K edges e_i at random and change it to another one randomly from our graph with a uniform proposal (not allowing edges which are already in the significant set of separating edges) either from the whole set of edges or from adjacent ones.

Step B′1b: We then separate the $K+1$ disconnected graphs which result from this, and calculate their means $\overline{\boldsymbol{x}}_i$ and their sizes n_i.

Step B′1c: We propose a new covariance matrix Σ' from a distribution

$$\Sigma|D, \boldsymbol{e}, \boldsymbol{\mu}_i = \overline{\boldsymbol{x}}_i \ \forall \ i \sim \text{InvWishart}\left(n+m, \Psi + \sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^T - \sum_{i=1}^{K+1} n_i \overline{\boldsymbol{x}}_i \overline{\boldsymbol{x}}_i^T\right),$$

which is the approximate posterior of Σ given the rest of the parameters, taking $\boldsymbol{\mu}_i = \overline{\boldsymbol{x}}_i$. See Appendix for the generation of Wishart distributed variables.

Step B′1d: We then propose $\boldsymbol{\mu}_i'$ from $\boldsymbol{\mu}_i|\Sigma, \boldsymbol{e}, D$ above. So here we have a Metropolis-Hastings step with an independent sampler for $\boldsymbol{e}$, an approximate Gibbs sampler for Σ and a Gibbs sampler for the $\boldsymbol{\mu}_i$'s.

Step B′1e: We need to accept or reject $\boldsymbol{e}$, Σ, $\boldsymbol{\mu}$. We have

$$\pi(\boldsymbol{e}', \Sigma', \boldsymbol{\mu}'|D) \propto f(D|\boldsymbol{e}', \Sigma', \boldsymbol{\mu}')p(\boldsymbol{\mu}')p(\Sigma')p(\boldsymbol{e}').$$

Therefore our acceptance ratio is given by

$$\begin{aligned} A' &= \frac{\pi(\boldsymbol{e}', \Sigma', \boldsymbol{\mu}'|D)q(\boldsymbol{\mu}' \to \boldsymbol{\mu})q(\Sigma' \to \Sigma)q(\boldsymbol{e}' \to \boldsymbol{e})}{\pi(\boldsymbol{e}, \Sigma, \boldsymbol{\mu}|D)q(\boldsymbol{\mu} \to \boldsymbol{\mu}')q(\Sigma \to \Sigma')q(\boldsymbol{e} \to \boldsymbol{e}')} \\ &= \frac{f(D|\boldsymbol{e}', \Sigma', \boldsymbol{\mu}')}{f(D|\boldsymbol{e}, \Sigma, \boldsymbol{\mu})} \frac{p(\boldsymbol{e}')}{p(\boldsymbol{e})} \frac{p(\Sigma')}{p(\Sigma)} \frac{p(\boldsymbol{\mu}')}{p(\boldsymbol{\mu})} \frac{\pi(\boldsymbol{\mu}|\Sigma, \boldsymbol{e})}{\pi(\boldsymbol{\mu}'|, \boldsymbol{e}', \Sigma')} \frac{\pi(\Sigma|D, \boldsymbol{e}, \overline{x}_1, \overline{x}_2)}{\pi(\Sigma'|D, \boldsymbol{e}', \overline{x}_1', \overline{x}_2')} \frac{p(\boldsymbol{e})}{p(\boldsymbol{e}')} \\ &= \frac{f(D|\boldsymbol{e}', \Sigma', \boldsymbol{\mu}')}{f(D|\boldsymbol{e}, \Sigma, \boldsymbol{\mu})} \frac{p(\Sigma')}{p(\Sigma)} \frac{p(\boldsymbol{\mu}')}{p(\boldsymbol{\mu})} \frac{\pi(\boldsymbol{\mu}|\Sigma, \boldsymbol{e})}{\pi(\boldsymbol{\mu}'|, \boldsymbol{e}', \Sigma')} \frac{\pi(\Sigma|D, \boldsymbol{e}, \overline{\boldsymbol{x}}_1, \overline{\boldsymbol{x}}_2)}{\pi(\Sigma'|D, \boldsymbol{e}', \overline{\boldsymbol{x}}_1', \overline{\boldsymbol{x}}_2')} \end{aligned}$$

Here

$$\begin{aligned} p(\boldsymbol{e}') &= \prod_{i=1}^{K} \frac{1}{n-i+1}, \\ p(\Sigma') &= \left(2^{md/2} \times \pi^{d(d-1)/4} \times \prod_{i=1}^{d} \Gamma\left(\frac{m+1-i}{2}\right)\right)^{-1} \\ &\quad \times |\Psi|^{m/2} \times |\Sigma|^{(d+m+1)/2} \times \exp\left(-\frac{1}{2} tr\left(\Psi\Sigma^{-1}\right)\right), \\ p(\boldsymbol{\mu}_i) &= (2\pi)^{-d/2} \times \left(\frac{|\Sigma|}{\tau_{prior}}\right)^{-1/2} \exp\left(-\frac{1}{2}\boldsymbol{\mu}_i^T \left(\frac{\Sigma}{\tau_{prior}}\right)^{-1} \boldsymbol{\mu}_i\right), \end{aligned}$$

and it can easily be calculated that

$$\begin{aligned} f(D|\mathbf{e}', \Sigma', \boldsymbol{\mu}') &= \prod_{i=1}^{n} (2\pi)^{-d/2} |\Sigma|^{-1/2} \exp\left(-\frac{1}{2}\left(\boldsymbol{x}_i - \boldsymbol{\mu}_{c_i}\right)^T \Sigma^{-1} \left(\boldsymbol{x}_i - \boldsymbol{\mu}_{c_i}\right)\right), \\ \pi(\boldsymbol{\mu}'|\boldsymbol{e}', \Sigma') &= \prod_{i=1}^{K+1} (2\pi)^{-d/2} \times \left(\frac{|\Sigma|}{n_i + \tau_{prior}}\right)^{-1/2} \times \\ &\quad \exp\left(-\frac{1}{2}\left(\boldsymbol{\mu}_i' - \frac{n_i \overline{\boldsymbol{x}}_i}{n_i + \tau_{prior}}\right)^T \left(\frac{\Sigma}{n_i + \tau_{prior}}\right)^{-1} \left(\boldsymbol{\mu}_i' - \frac{n_i \overline{\boldsymbol{x}}_i}{n_i + \tau_{prior}}\right)\right), \\ \pi(\Sigma'|\boldsymbol{e}', D, \overline{\boldsymbol{x}}') &= \left(2^{(n+m)d/2} \times \pi^{d(d-1)/4} \prod_{i=1}^{d} \Gamma\left(\frac{n+m+1-i}{2}\right)\right)^{-1} \\ &\quad \times |S|^{(n+m)/2} \times |\Sigma|^{-(n+m+d+1)/2} \times \exp\left(-\frac{1}{2} tr\left(S\Sigma^{-1}\right)\right), \text{ where} \\ S &= \Psi + \sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^T - \sum_{i=1}^{K+1} n_i \overline{\boldsymbol{x}}_i \overline{\boldsymbol{x}}_i^T. \end{aligned}$$

So,we calculate the ratio A' and accept the step with probability

$$\min(1, A').$$

If we accept, we set

$$\begin{aligned} \boldsymbol{e}^{(t+1)} &= \boldsymbol{e}' \\ \Sigma'' &= \Sigma' \\ \boldsymbol{\mu}'' &= \boldsymbol{\mu}' \end{aligned}$$

otherwise we set

$$\begin{aligned} \boldsymbol{e}^{(t+1)} &= \boldsymbol{e}^{(t)} \\ \Sigma'' &= \Sigma_t \\ \boldsymbol{\mu}'' &= \boldsymbol{\mu}^{(t)} \end{aligned}$$

Step B′2: We then generate

$$\Sigma^{(t+1)}$$

from the posterior conditional

$$\Sigma | D, \boldsymbol{e}^{(t+1)}, \boldsymbol{\mu}''$$

(and accept with probability 1 as in standard Gibbs sampler).

Step B′3: Lastly, same as step 2, we generate $\boldsymbol{\mu}^{(t+1)}$ from

$$\mu_i | D, \boldsymbol{e}^{(t+1)}, \Sigma^{(t+1)}.$$

We now go back to step 1.

As above, this chain is irreducible and aperiodic.

3.2.4 The label-switching problem

In order to be able to draw conclusions about the distribution of individual cluster means, we need to have an efficient method of labelling the clusters each time, otherwise the cluster parameters may be falsely allocated to the wrong cluster.

We use the algorithm described in Stephens (2000):
Starting with some initial values for the permutations $\nu_1, \ldots, \nu_N$ of the first N steps (setting them all to the identity permutation for example), iterate the following steps until a fixed point is reached:

Step 1: Choose $\hat{a}$ to minimise $\sum_{t=1}^{N} \mathcal{L}_0\left(\hat{a}; \nu^{(t)}\left(\theta^{(t)}\right)\right)$.

Step 2: For $t = 1, \ldots, N$ choose ν_t to minimise $\sum_{t=1}^{N} \mathcal{L}_0\left(\hat{a}; \nu^{(t)}\left(\theta^{(t)}\right)\right)$.

Using the loss function suggested in Stephens (2000), Starting with some initial values (the identity permutation, say), the algorithm becomes a single step:

Pick a labelling to maximise

$$\sum_{j=0}^{n} \log \frac{1}{t} \left(\sum_{i=1}^{t} \mathbb{I}_{c_j^{(i)} = c_j^{(t)}} \right)$$

where c_j denotes the cluster of node j. In other words, this algorithm picks labels for each group so that "as many nodes as possible belong to their favourite group so far".

3.3 Missing nodes

Here we look at data where one or more nodes are not observed in our sample. It is very unlikely that a real set of data will contain all haplotypes in the tree. Now we consider a set of sequences not defining a complete tree, but still with no homoplasy or recombination. We follow the algorithm described below to work out the topology of the complete true mutational tree.

1. We calculate the number of disconnected groups of nodes. For every pair of groups, we find the closest distance between two nodes belonging to each group. We will refer to these pairs of nodes as the representatives between two groups. In this case because no homoplasy is present, these are unique for each pair of groups.

2. Then we find the minimum of these minimum distances, and set one of the two nodes to be the reference node (without loss of generality either can be equally chosen to be the reference node).

3. We then find all the pairs of groups which have the reference node as one of their two representatives. We store the separating mutation positions between each one of these representatives and the reference node.

4. Then we find the separating mutation(s) which occurs most times, and we pick one of them, which we call the "reference mutation" . This mutation has to be the one that happened closest to the reference node, and so we create an extra node which is identical to the reference node except at the reference mutation position. We then go back to step 1 and repeat until the whole graph is connected.

The above algorithm works perfectly when there is no homoplasy in our data. In this case, the representatives between two groups of nodes are unique and that if the representative of groups i and j are nodes x and y, and the representatives of groups i and k are nodes x and z, then the representatives between groups j and k are nodes y and z.

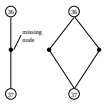


Figure 3.2: A missing node of degree 2

Once we have a complete connected tree, we carry out the MCMC algorithm described in Section 3.2.1. The only complication is that we have to make sure that whichever two edges we pick as causal ones, we should make sure they split our nodes into $K+1$ groups, none of which has no data points. This could happen if we had a missing node of degree 2, and we proposed to delete the two edges connected to it (see Figure 3.2).

In a situation like Figure 3.2, proposing to remove either of the two edges will have exactly the same effect. In the final histogram, they should have almost identical probabilities, and we need to add them up and consider them as 1 edge possibility. This only happens in the cases where a missing node is not deterministic, which implies that neither are the two (or more) edges, as there is a 50-50 chance they will be in one order or the other (and accordingly if more than 2 edges). In cases of deterministic nodes and edges, each edge should be treated separately independent of whether it's adjacent to missing or observed nodes.

3.3.1 Example: Simulated dataset

Using the same sequences generated in the previous example but removing nodes 5 and 1 the program produces the same tree as the true one, and then draws the same conclusions as before (with a slight difference as we have removed a few data points).

Using the reduced graph from our earlier example (see Figure 3.3), we follow the steps described above in order to complete the two missing nodes. Our algorithm first finds the closest nodes. In this case, nodes 4 and 8, 0 and 4, 0 and 8 are a distance 2 apart. Our program arbitrarily picks 0 and 4, and sets 0 to be the "reference" node. Then we find all other representatives of groups which are closest to 0, namely 4,14,25,39,8. Since 0 and 4 are a distance 2 apart, we have 2 choices for the intermediate node. However, looking closely at which nodes include which mutation, the program finds that one of the two appears in ALL of the other relevant representatives, whereas

the second one only in node 4. Hence it chooses the most occurring mutation, Thus creating the top missing node.

Then we find again the two closest nodes, here (arbitrarily as there is more than one equivalent pairs) we pick 5 and 14. As before, we set 5 to be the reference node, and then 14, 25, 39 are the 3 other representatives. We have 2 choices for the intermediate node, but one of the mutations appears once whereas the other one 3 times, hence we pick the latter and form the lower missing node.

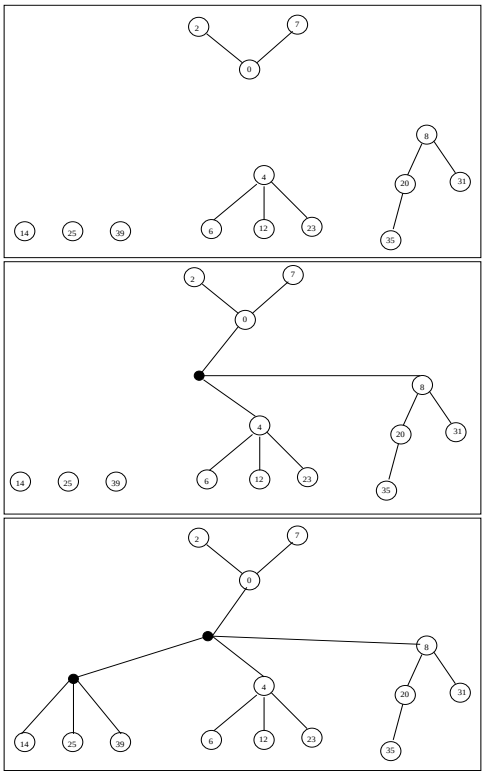


Figure 3.3: The figure above shows the 3 iterations of finding the two missing nodes, using the reduced graph from example 2.1

3.4 Homoplasy

Homoplasy has the effect in our network of creating two different edges representing a mutation on the same restriction site. In some cases, loops may be formed. However, even in the unlikely event of homoplasy, a loop is extremely unlikely. A loop always has to be of an even number of edges, since the only way we can get from a haplotype back to itself through mutations is by doing and undoing mutations. Of these edges, it is much more likely that one did not actually occur. The sequence of haplotypes would still be the same, only with two haplotypes which seem one mutation apart not actually being one mutation apart.

3.4.1 Homoplasy and missing nodes

One of the main problems of homoplasy is that it makes the algorithm described above which determines missing nodes not well-defined. If no homoplasy is present, then two sequences that are x mutations apart will also be x positions apart in the cladogram. However, in the presence of homoplasy this is not always true.

In forming an algorithm to deal with such cases, we need to first decide what is more important: To form a cladogram in which sequences are apart as little as possible, or to form the smallest cladogram possible? In the first case, too many missing nodes may need to be determined and inserted. In the second case, we insert the fewest possible missing nodes, but some nodes will be too far apart in the cladogram compared to their actual sequence distance.

It is also possible that in some cases there are more than one pair of nodes being the closest between two groups of nodes, however this is rare. In that case we need to insert the missing nodes step in our MCMC iterations, and we have to pick one of the two pairs at random.

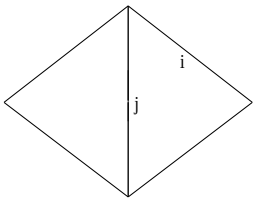
3.4.2 Homoplasy and loops

If loops are present in our cladogram, we know that it is extremely unlikely that all these mutations happened. However, it is not possible to identify which of the mutations of our loops did not actually occur. We try to estimate the probability of specific mutations (e.g. A to G etc.), and we know that transitions are much more likely than transversions. Once we have done that, we have a measure of the likelihood of every edge, and thus we can include one more step in our MCMC algorithm which first decides one which tree topology to look at.

3.4.3 Comments

One of the questions arising here is what to do with edges belonging to more than one loop. According to the above method, that edge would have double the probability of being deleted, once for the breaking of each of the two loops it belongs to. However, we want each possible loop-free tree to have equal probability, so whenever 2 (or more) loops are dependent, we want to choose 2 edges simultaneously with equal probability so that the remaining graph is a tree. To do so, we delete an edge from each loop (with replacement), and if any two coincide, we pick different ones afresh. We can check that this gives us equal probability of any pair of edges being deleted in a simple example (see Figure 3.4.3).

We have $\binom{5}{2}$-2 possible pairs of edges, and we want all of them to have equal probability, ie 1/8. Using the proposal described above the probability is:



$$\mathbb{P}(\text{edges i,j}|\text{accepted}) = \frac{\mathbb{P}(\text{edges i,j} \cap \text{succeed})}{\mathbb{P}(\text{succeed})} = \frac{9}{8}\frac{1}{9} = \frac{1}{8}$$

and so we preserve the equal probability of every pair of edges.

The method of removing loop at each step produces a complication: if the true causal edge is part of a loop, then a lot of the time we don't have the choice to pick it as a proposed causal edge, as it gets deleted in the earlier stage which creates a loop-free graph. For that reason, in order to examine the significance of loop edges, we need to look at the distribution of the edge in question conditional on that edge not having been deleted at the previous stage.

Another consequence of homoplasy is that when we propose an edge e to be the causal one, it is possible that that specific mutation happens somewhere else in our network too. Then, if we are dealing with phenotypic data, we should automatically split the graph up to as many subgraphs formed when deleting any edge representing a mutation at the specific restriction site, since we are assuming that restriction sites are independent, i.e. that a mutation is either causal or not independent of other mutations present. In these cases, one (or more) extra clusters are formed, and we assign means to them randomly from available means.

If we are dealing with phylogeographic data, then we don't simultaneously delete a mutation on a certain restriction site across the whole graph, because we are not looking for any mutation in that restriction site to have caused a significant geographic event, but that specific mutation we have picked which happened at a specific time and place.

If we assume that mutations at different sites are independent of each other then we should delete every edge representing that site at once, and then assign extra clusters to have any of the $K + 1$ means at random. However, if we assume some dependence between different sites, then it could be that a mutation had an effect only at one of the 2 times it appears, because it is combined with some other change in the sequences, and assuming independence would lose that information.

3.4.4 Algorithm

The algorithm in this case is similar to Steps A1-A4, only this time we also have to pick a tree topology at each cycle. Since clusterings are only defined within a tree topology and a certain clustering can be impossible under a different tree topology, we update the separating edges and tree topology together.

Step C1: Generate a new root from the estimated distribution in (3.1) and accept with probability A_r.

Step C2: Update the amino acid frequencies $\boldsymbol{\pi}$ and transition rates $\boldsymbol{\alpha}$ as in steps A2-A4.

Step C3a: Generate a new tree topology from the conditional posterior, and calculate its acceptance ratio A_5.

Step C3b: Continue with the phenotypic analysis steps B1-B3, replacing the accept-reject probability with $A \times A_4$, and if we accept replace with the new values both for the separating edges and the parameters of the clusters as well as the tree topology. The only difference is that if we propose a tree topology which doesn't include all the separating edges from the previous cycle or a set of separating edges which result in $n_i = 0$ for some i, we reject that move straight away.

It can easily be checked that despite the fact that in this case the acceptance probability is often equal to zero, the chain remains irreducible (and aperiodic).

3.4.5 Example: The beetle data

We use the beetle mitochondrial sequence data from 134 individuals, yielding 69 haplotypes. Since we are looking at mitochondrial sequence, no recombination is possible as the DNA is passed only from one of the parents.

Of the total 570 sites, only 66 were variable. The output graph is given below, again using TCS1.21 for a graphical representation of the output of our program. The numbers correspond to the haplotypes found in the localities given in the Appendix.

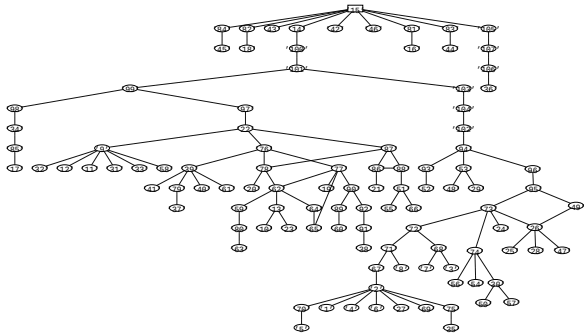


Figure 3.4: Graph formed using the Beetle data from La Palma

There are 3 loops formed, some of which share common edges. Here haplotypes 70 and above are inferred ones and not observed. The true cladogram is assumed to be a subset of the cladogram in the above figure, after removing loops.

3.5 Phylogeographic and phenotypic analyses for an unknown number of separating edges

In practice, we almost never know the true number K of underlying significant edges. We therefore also have to make inference about the number of significant clusters we are trying to split our data into. To do so, we use a Reversible-Jump MCMC method similar to the one described in Richardson and Green (1997), which allows jumping between parameter spaces with different size:

Step D1a: After running a cycle of steps C1-C5, we then decide with probability p_{split} to split one of the existing clusters into 2 so that $K^{(t+1)} = K^{(t)} + 1$, and with probability $p_{merge} = 1 - p_{split}$ to combine 2 of the existing clusters into 1 so that $K^{(t+1)} = K^{(t)} - 1$.

Step D1b: a) If we decide to merge 2 clusters, we pick at random a pair of adjacent (in terms of the cladogram) clusters i, j (i.e. one of the existing significant edges which connects 2 clusters) and we merge them, removing that edge e_i from the vector $\boldsymbol{e}$ of separating edges. We calculate the resulting sample mean and propose a new mean for this merged cluster formed from a distribution

$$\boldsymbol{\mu}_i \sim \text{MVN}\left(\frac{n_i\overline{\boldsymbol{x}}_i}{n_i+\tau_{prior}}, \frac{1}{n_i+\tau_{prior}}\Sigma\right)$$

So we propose to move from $(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_{K^{(t)}+1})$ to $(\boldsymbol{\mu}'_1, \ldots, \boldsymbol{\mu}'_{K^{(t)}})$, according to the following transformation:

$$f((\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_{K^{(t)}+1}, \boldsymbol{\mu'}_c) = (\boldsymbol{\mu}'_1, \ldots, \boldsymbol{\mu}'_{K^{(t)}}, \boldsymbol{\mu}_i, \boldsymbol{\mu}_j),$$

where the new means have all stayed the same (with rearranging of their order) apart from the one belonging to the merged cluster c.

b) If we decide to split one of the existing $K_t + 1$ clusters, say cluster c, we choose one of the edges within that cluster and remove it, increasing the size of $\boldsymbol{e}$ by 1. We then calculate the resulting sample means and propose new $\boldsymbol{\mu}_{i,j}$ as usual from the posterior distribution.

Step D1c: Then the acceptance probability of a merging move becomes $\alpha = \min(1, A_5)$ where

$$A_5 = \frac{f(D|\boldsymbol{\mu'})p(\boldsymbol{\mu'_g})p(\boldsymbol{e'})q(\boldsymbol{\mu}'_g \to (\boldsymbol{\mu_i}, \boldsymbol{\mu_j}))q(\boldsymbol{e'} \to \boldsymbol{e})}{f(D|\boldsymbol{\mu})p(\boldsymbol{\mu_i})p(\boldsymbol{\mu_j})p(\boldsymbol{e})q((\boldsymbol{\mu_i}, \boldsymbol{\mu_j}) \to \boldsymbol{\mu}'_g)q(\boldsymbol{e} \to \boldsymbol{e'})}\,|J|$$

where $f(D|\boldsymbol{\mu})$, $p(\boldsymbol{\mu'_i})$ are given by the usual formulae (see section bla), $q((\boldsymbol{\mu_i}, \boldsymbol{\mu_j}) \to \boldsymbol{\mu}'_g)$ is just the posterior of $\boldsymbol{\mu}'_i|D, \boldsymbol{e}', \Sigma$,

$$\begin{aligned}
p(\boldsymbol{e}) &= \frac{1}{\binom{\text{total \# of edges}}{K^{(t)}}} \\
q(\boldsymbol{e} \to \boldsymbol{e'}) &= \frac{1}{K^{(t)}} && \text{for a merging move, and} \\
q(\boldsymbol{e} \to \boldsymbol{e'}) &= \frac{1}{(K^{(t)}+1) \times (\text{\# of edges in cluster c})} && \text{for a splitting move,}
\end{aligned}$$

and J is the Jacobian of the square map

$$f : \mathbb{R}^{K^{(t)}+2} \longrightarrow \mathbb{R}^{K^{(t)}+2},$$

so that $J_{ij} = \frac{\partial f_i}{\partial y_j}$. Clearly, the Jacobian in this case is 1.

Similarly, the acceptance probability of a splitting move becomes $\alpha = \min(1, A_4^{-1})$. We decide to accept or reject the proposed move, with some terms replaced appropriately.

So, a run through steps C1-C5 and D1 produces a chain with a stationary distribution for the number of separating edges K, the specific set of significant edges $\boldsymbol{e}$ and means of the clusters $\boldsymbol{\mu}$ as well as the rest of the parameters.

3.5.1 Example: The beetle data

We apply the method described above to the beetle data, assuming appropriate parameters for the priors and distributions results shown in Table 3.5.1. Increasing the range for the allowed number of mutations we get a decreasing posterior for the number of mutations being ≥ 4. The significant clusters largely agree with the clusters deduced in Emerson's paper. The three main clusters shown in Figure 2.1 are the ones that our analysis also gave.

Running the chain twice using 3 and 4 significant mutations respectively (fixed throughout the program), we obtain the following graphs 3.5, 3.6. The 4 (or 5) colours correspond to the 4 (or 5) significant clusters. On the left we show the distribution of clusters in the 18 locations based on our method and on the right based on a usual clustering algorithm not taking the sequences into account. Clearly, with both 3 and 4 separating mutations, the results are much more significant in the case of clustering within the cladogram. Indeed we confirm that the separating volcano meant that most probably no individuals migrated from a population just above it to one just below it

group of edges	$\mathbb{P}(sig\|D)$
22-97	0.1723
97-99	
94-102	0.0898
101-103	
102-104	
103-104	
14-15	0.0439
100-101	0.0409
14-100	
9-22	0.0409
15-105	0.0218
105-107	
106-107	
36-106	
94-96	0.0216
95-96	

node	deg	$\mathbb{P}(\text{root}\|D)$
15	9	0.2909
2	8	0.3704
9	7	0.1210
26	5	0.0442
39	5	0.0244
73	5	0.0391
# mutns	post prob	
2	0.1545	
3	0.3300	
4	0.3400	
5	0.1754	
acceptance ratio 0.2588		
27 mins approximately		

Table 3.1: Output of the program for the beetle data

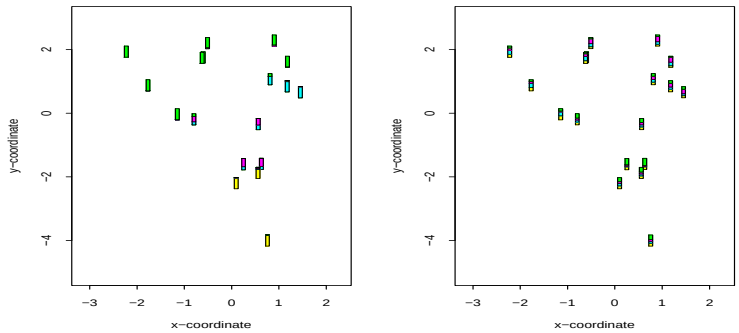


Figure 3.5: Graph showing the distribution of clusters in each location for 3 mutations

(or vice versa). The same is true with the two clusters on the east of the island, where the clusters are strongly separated.

The significance of edges is very strongly correlated between them. For that reason, in order to infer the most likely set of 3 or 4 edges to be the truly significant one, we need to look at the matrix of correlations. Doing so, we obtain the mode significant set of edges which is (22-97,94-102, 73-74, 72-73). We plot the same graph for the mode clustering in Figure 3.7 and we notice that this is almost identical to the clustering derived by Emerson et al. (2005).

In the case of 3 significant mutations we get the following means on the x and y-coordinates in Fig. 3.8. In Figure 3.9 of the estimated densities, the bimodality shows that there still appears to be a label-switching problem, which cannot be totally avoided.

3.5.2 Example: Simulation using derived posterior

We now use the results we obtained above to simulate a new set of data and check if our method indeed draws the same conclusions. We pick the state at which our chain is at the mode clustering and use the tree topology $T^{(t)}$, significant separating edge set $\boldsymbol{e}^{(t)}$ (the mode), covariance matrix $\Sigma^{(t)}$, vector of means $\boldsymbol{\mu}^{(t)}$, transition probability matrix $P^{(t)} = p_{ij^{(t)}}$, gene frequencies $(\pi_A,\ \pi_G,\ \pi_T,\ \pi_C)^{(t)}$ and the root $r^{(t)}$ and we simulate the sequences down the given tree and then we simulated the bivariate normal locations with the given means and covariance matrix given the number of data point from each haplotype. Our program pick ups the correct edges as significant, as we would expect.

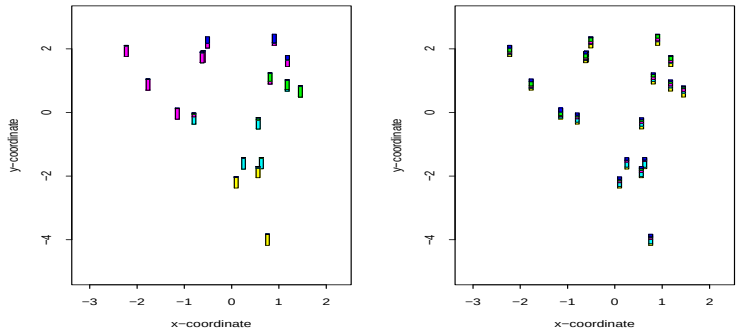


Figure 3.6: Graph showing the distribution of clusters in each location for 4 mutations

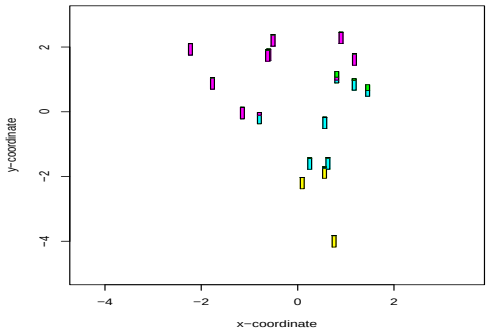


Figure 3.7: Graph showing the distribution for the mode clustering

3.6 Recombination

Often, homoplasy events may be explained by a recombination event. In addition, in the event that one branch is connected to the rest of the network through a series of missing nodes, it is possible that in fact recombination occurred. Recombination has not been implemented yet, but a procedure is described here.

In both cases, we try to find possible recombination events. This means that for a sequence S_1 we find two sequences S_2 and S_3, and a letter $0 < k < l$ where l is the length of the sequences, such that S_1 is identical to S_2 up to letter k and identical to S_3 from $k+1$ to l. Usually more than one possibilities are found. In the same manner as in deciding on a tree topology in the case of homoplasy, we can also pick any of these possibilities with equal probability. We also assign a probability to recombination having occurred or not compared to homoplasy and a series of nodes missing. We use the convention that we assume recombination is it explains 2 or more homoplasies (see Aquadro et al. (1986)). A series of missing nodes induces more homoplasies, as does a loop.

Once we have done that, we describe the way to analyse a network including recombinations. A recombinant haplotype is joint to two different haplotypes, but these edges may only be deleted simultaneously. In addition, if some other edge is proposed to be the causal one, then the recombinant branch will belong to a cluster if it carries the significant mutation that the cluster carries. It is possible that a branch belongs to more than one clusters, and in that case we include it in the calculation of the likelihoods etc. in both clusters, which is important for the estimates of the variance.

For example, if we have proposed to delete two edges, one being the mutation at position 3, where on one side the haplotypes have a T and on the other an A, and the other one at position 5, where we get an A and a G on either side, then the recombinant branch, which has an A on both position 3 and 5, will belong to the cluster that has an A at the 3rd position, and the cluster that has an A at the 5th position. If this is the same cluster, then the recombinant branch only belongs to one cluster.

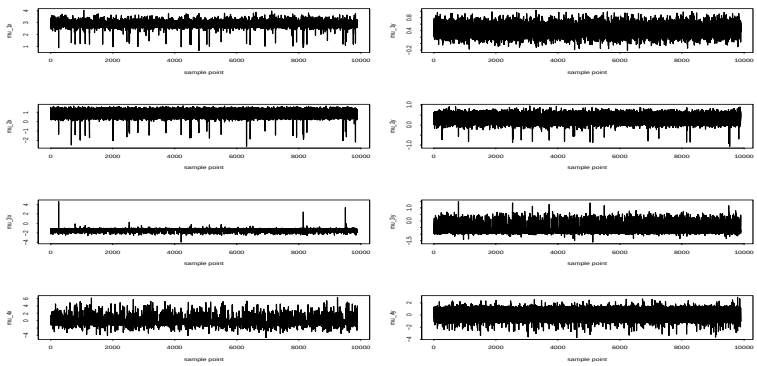


Figure 3.8: Trace plots of the means of the x and y coordinates in the 4 clusters

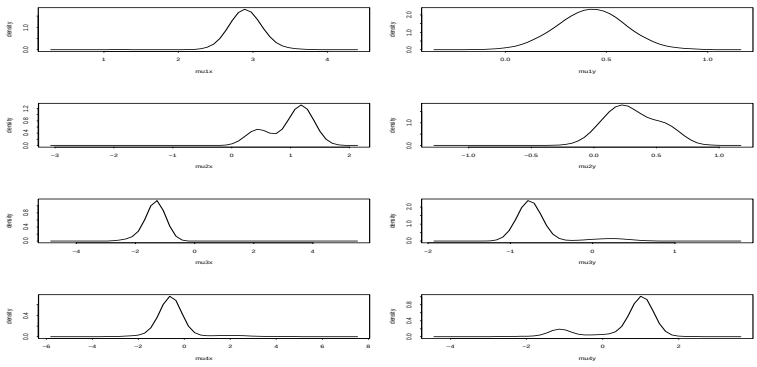


Figure 3.9: Estimated density of the means of the x and y coordinates in the 4 clusters

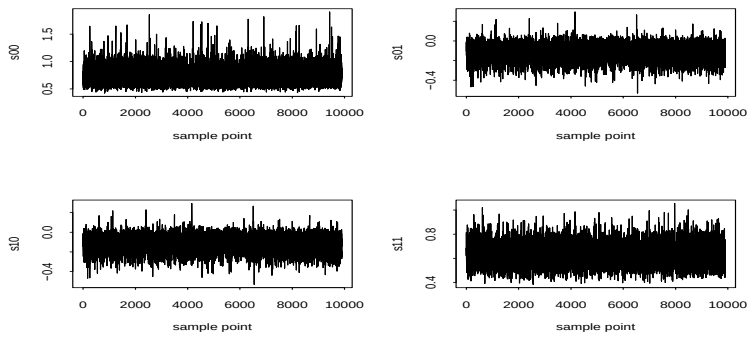


Figure 3.10: Trace plots of the covariance matrix elements

Chapter 4

Conclusion

In this essay we began by presenting existing methods of carrying out phenotypic and phylogeographic analyses. The main method used at the moment, Nested Clade Analyses, is based on deciding on a unique (or nearly unique) cladogram in order to analyse the phenotypic data. The analysis is highly dependent on the initial tree inferred, and the criteria proposed used are not always well-defined, and can be subjective. A wrong choice of tree can lead to very different conclusions about the data. Moreover, it is difficult to incorporate more parameters into the analysis, if the data is mutli-dimensional.

Bayesian phylogenetic analyses aim at inferring the coalescent tree behind a number of taxa. They are oriented towards estimating the Time to the Most Recent Common Ancestor rather than the number of mutations involved. They are most powerful for a relatively small number of taxa, and are generally time consuming (it might take weeks or months for the MC chain to converge) since the order in which mutations and coalescence events occur (and their respective times) is important. Hence phylogenetic trees are not always appropriate for phenotypic analyses.

Here we take an approach to Bayesian cladograms. We address some of the common problems when trying to infer a cladogram, and propose methods to overcome them. The advantage of our methods is that the danger of choosing the wrong tree is avoided, since all trees are taken into consideration simultaneously, and investigated in relation to the geographical/phenotypic data. Although in the beetle case only the geographical position is given, more parameters could have been incorporated in our study.

We take a clustering approach to this problem, so our method decides on the best way to split our data into significantly different groups. We suggest an algorithm which doesn't assume a fixed number of clusters, using Reversible Jump Markov Chain Monte Carlo methods. We also propose a solution to the label-switching problem, to show a significant improvement in the interpretation of the output. We implement our methods for the beetle data.

Throughout the study, we have assumed that the Markov Process for mutations is the same across all sites. However, this is not always true, and we should use different transition matrices for different restriction sites. A way in which this can be done is to introduce an extra parameter θ which is a measure of the mutation rate, and to pick a different θ for each codon† position using a Gamma distribution. That is, we assume that the Q-matrix is given by θQ so that the stationary distribution remains the same. We also propose a way to deal with recombinations. Both of these are yet to be implemented.

A very similar analysis as we described in this study can be applied for data which is not in $\mathbb{R}$, using for example logistic regression in the case of a 0/1 characteristic trait (e.g. the presence or not of a disease), and updating the parameters in the MCMC steps.

Appendix 1

Glossary

- **Aligned DNA Sequences:** A set of DNA sequences (including gaps) which are arranged so that for each sequences the amino-acid positions are matched. This means that, for example, if in two sequences a gap is matched with an amino-acid, this indicates that a deletion (or insertion) has occurred.
- **Amino-acid:** In DNA, adenine, guanine, thymine or cytosine, represented by A, G, T, C respectively.
- **Binary tree:** A tree is a graph with no loops (cycles). A binary tree is a tree where all internal nodes have order 3, the root has order 2 and leaves order 1.
- A **cladogram** is similar to an unrooted phylogenetic tree, where every edge represents a single mutation (see Fig. 1). All edges of a cladogram have the same length. In the example below, haplotypes 1,2,3,4,5 correspond to the following sequences:

Haplo	Sequence
1	CAAAAAAACC
2	AAAAAAAAAA
3	AACAAACCCC
4	AAACAACCCC
5	ACAACCACCC

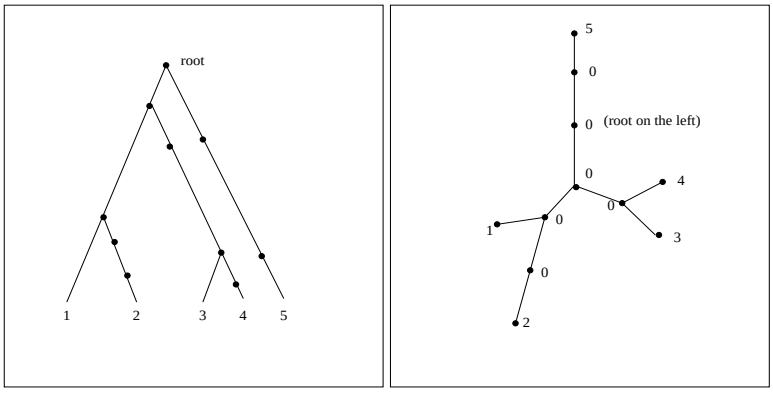


Figure 1.1: An example of a phylogenetic tree and a simple cladogram

In the phylogenetic tree, nodes represent mutations. At each coalescence event, the mutation only happens for the left (or right) clade, so that they separate. Note that the phylogenetic tree is not unique for this cladogram, and if we don't know the exact nature of each coalescence, we also don't have uniqueness vice versa. Also, although in the cladogram the zero nodes don't have a unique arrangement, this is not crucial to our study, as they don't enter the NCA.

- **Coalescence:** Uniting two genes into one, through their MRCA
- **Codon:** A sequence of 3 amino acids, which work as a unit in DNA and RNA functions, i.e. in terms of DNA replication (copying sequences).
- **Fragmentation:** When a population is broken into two or more subpopulations, e.g. by the formation of a canyon
- **Genetic Locus:** The position on a chromosome of a gene or other chromosome marker; also, the DNA at that position. The use of locus is sometimes restricted to mean regions of DNA that are expressed. The specific physical location of a gene on a chromosome. A stretch of DNA at a particular place on a particular chromosome – often used for a 'gene' in the broad sense, meaning a stretch of DNA being analysed for variability (e.g., a micro satellite locus).
- **Genetic Marker:** A gene or other fragment of DNA whose location in the genome is known.
- **Haplotype:** A set of closely linked genetic markers present on one chromosome which tend to be inherited together. We can either treat each individual letter of a sequence as a haplotype, or treat the whole sequence as a haplotype. The distinction is whether it has some validity to consider the whole sequence as one item which behaves as a whole, or whether each letter may be treated as a separate entity (although they might still be in linkage disequilibrium to some degree).
- **Homoplasy:** When one or more mutations to have occurred in the same restriction site, leading to parallelisms (e.g. AGC, AGT, AGA are all the same distance apart) or convergence of patterns (e.g. getting the same haplotype or part of haplotype by two different routes).
- **Linkage Disequilibrium:** When nucleotides in different sites are not inherited and do not appear independent of one another, which results to the observed frequencies to not be the ones expected by independence.
- **Molecular clock**: the hypothesis that nucleotide or amino acid substitutions occur at more or less fixed rate over evolutionary time, like the slow ticking of a clock. Figure 2 shows an example of a graph with and without a molecular clock. If a molecular clock is assumed, the pair of distances to the adjacent leaves are equal for each internal node.

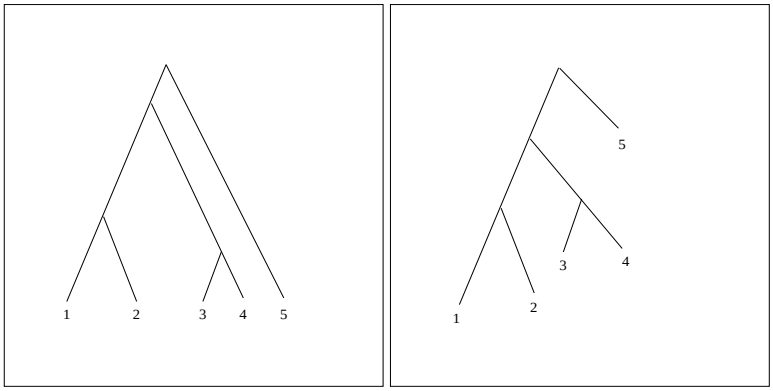
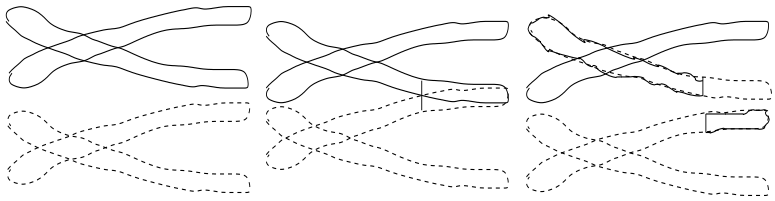


Figure 1.2: An example of a phylogenetic tree with and without a molecular clock

- **Nucleotide:** A subunit of DNA or RNA consisting of a nitrogenous base (adenine, guanine, thymine, or cytosine in DNA; adenine, guanine, uracil, or cytosine in RNA), a phosphate molecule, and a sugar molecule (deoxyribose in DNA and ribose in RNA). Thousands of nucleotides are linked to form a DNA or RNA molecule.
- **Phenotype vs. Genotype**: The genotype is the specific haplotype sequences governing some individual trait, whereas the phenotype is what the effect of that specific gene combination is on the trait.

- A **Phylogenetic Tree** is a tree showing the evolutionary interrelationships among various species or individuals (that are believed to have a common ancestor). In a phylogenetic tree, each node with descendants represents the most recent common ancestor of the descendants, and edge lengths correspond to time estimates.
- **Phylogeography:** The attempt to take into account the geographic distribution of species in establishing their phylogeny, and to understand the geographic patterns that may result from divergence, ultimately leading to speciation.
- **Recombination:** when two genes exchange a chunk of their DNA sequence, see below: Each of the strands shown above represents a chromosome which contains a DNA sequence,



so that when the chromosomes cross over and exchange a section of their DNA, the same happens to the two DNA sequences.

- **Restricted gene flow:** When a population shows limited movement so that genes stay within that population. Isolation-by-distance is one of the main reasons why this happens.
- **Restriction site:** A specific amino-acid position, or a chunk of the DNA of an individual which we are studying. Usually there is prior evidence to suggest that the mutation(s) we are looking for are in a specific locus of the DNA sequence, and so we only look at a portion of the whole genome.
- **SNP:** Single Nucleotide Polymorphism. DNA sequence variations that occur when a single nucleotide (A, T, C, or G) in the genome sequence is altered. Each individual has many single nucleotide polymorphisms that together create a unique DNA pattern.
- **Nucleotide Substitution:**When a base nucleotide changes (mutates) to a different one, e.g. a A becomes a T.
- **Taxon:** A category of the phylogenetic structure; e.g. when we are trying to find out the evolutionary history of primates, each primate represents a taxon. In a phylogenetic tree the taxa are simply the leaves of the tree.
- **Transition vs. transversion mutations:** Transitions are mutations changing A to G or C to T (and vice versa). Transversions are the remaining mutations. They are called this because transition mutations do not alter the chemical nature of the base, and they are often much more frequent.

Appendix 2

The Beetle data

Emerson et al. (2005) use the geologically young and well-characterised island of La Palma to generate phylogeographic predictions for Brachyderes Rugatus Rugatus, a curculionid beetle species occurring throughout the island in the forests of Pinus Canariensis. We have a sample of 135 beetles from 18 localities across the distribution of B. R. Rugatus for the construction of a haplotype network and application of nested clade phylogeographic analysis (NCPA) for 570 base pairs (bp) of sequence data for the mitochondrial DNA (mtDNA) cytochrome oxidae II (COII) gene.

Below is a map of the localities, and also the sampling localities, number of haplotypes in each locality, and haplotype numbers in each locality, with the number of individuals containing that haplotype in brackets if more than one. Haplotype numbers correspond to those in Fig. 3.4.5.

Locality
Number of Haplotypes
Haplotype (number of individuals)

Above Fuente de Olen
8
50, 51, 52, 53, 54, 55, 56, 57

Montaña Tagoja
4
30(4), 47(2), 48, 49

Fuente de Olen
7
24, 25, 26(2), 27, 28, 29, 30

El Bejenado
7
2(2), 3, 4, 5, 6, 7, 8

Aridane
6
2(3), 51, 66, 67, 68, 6

Montaña de la Venta
1
2(8)

El Jable
2
1(2), 2(8)

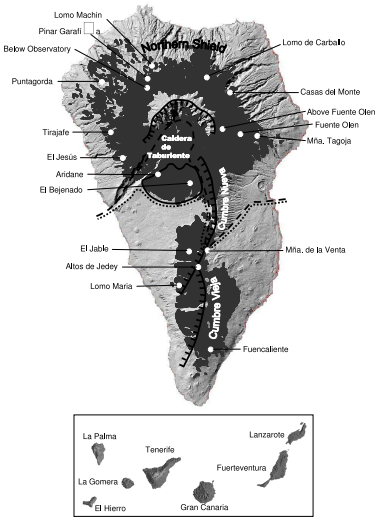


Figure 2.1: Map of locations

Lomo Maria
5
14(2), 15(3), 16, 17, 18

Fuencaliente
5
42, 43(2), 44(2), 45(2), 46

Altos de Jedey
4
15, 34(5), 35(1), 36(1)

Casas del Monte
2
9, 13

Lomo Carballo
4
9(4), 31, 32, 33(2)

Lomo Machin
5
9(4), 10, 11, 12, 13

Pinar Garafía
5
37, 38, 39(3), 40(2), 41

Below Observatory
5
13(2), 39(3), 58, 59, 60

Puntagorda
6
13, 19(2), 20, 21(2), 22, 23

Tirajafe
5
19, 38(3), 61(2), 62, 63

El Jesús
2
64(7), 65

Appendix 3

Overview of basic genetics

A Introduction

The idea in DNA sequence analyses is that we use sections from the DNA sequences of a number of individuals to draw various conclusions. The DNA sequence consists of the letters A,T,C,G, representing 4 amino acids. The complete DNA sequence is usually very very long (billions of letters) but in these analyses usually we only look at a section of it, which is from a few hundred to a few thousand letters long.

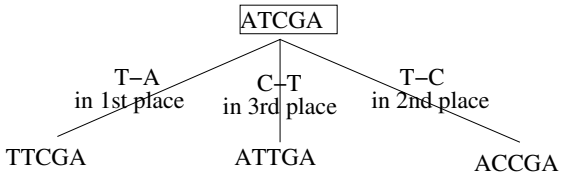
The DNA sequence is wrapped up chromosomes (23 pairs in humans, from mother and father) which store millions of amino acid pairs each. The sequences can be quite different from individual to individual, and not even have the same length, so it's not clear which pairs are "corresponding" between two individuals. Here we assume that the DNA sequence under analysis are actually aligned. That means that they have been matched letter by letter to satisfy a 1-1 correspondence.

B Cladograms

The purpose of a cladogram is to identify the evolutionary history of a DNA sequence. Say for example we have a set of data

ATCGA
ATTGA
ACCGA
TTCGA

We might conclude that the true underlying rooted cladogram is:



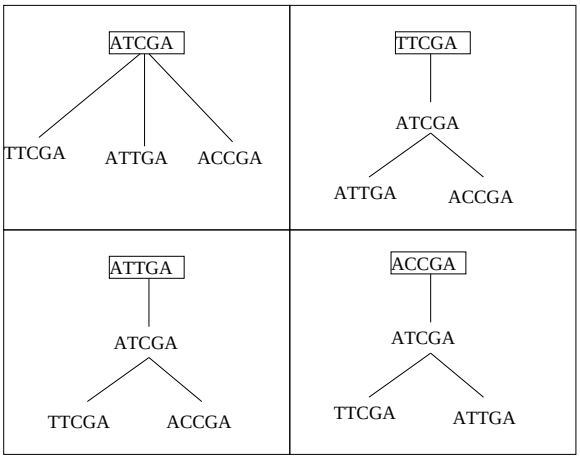
What this means is that initially ATCGA was the only 1 of the 4 DNA sequences present. For generations individuals with that DNA sequence were breeding and with the exact same sequence being inherited. The at some point someone's child was a mutant, yielding one of the other 3 DNA sequences (it's not possible from a cladogram to identify which one). So, a rooted cladogram only includes a measure of time in terms of mutation time units. Here the square represents the root of the cladogram. Often a cladogram is not rooted, so we have many possibilities:

So, each node in the cladogram represents a DNA sequence which for generations is inherited as is. A line connecting it to another node represents a mutation which gave a younger DNA sequence. In this context younger means that the first time this sequence appeared was later.

A mutation doesn't mean that the older sequence dies, however. Even though one individual gave birth to a mutant, there were plenty of other individuals with the same DNA sequence whose descendants had the identical DNA sequence.

A typical cladogram is not actually rooted. That means that the edges are not directed, i.e. we can't tell which way around the mutations happened. A general rule, however, is that nodes with higher degree (i.e. with lots of nodes being joint to them) represent older rather than younger DNA sequences. This is because a DNA sequence that has existed for millions of years is also much more likely to have mutated lots of times compared to one which has only been around a few hundred years.

In a typical dataset, we can't form a complete cladogram purely using the observed sequences as nodes, and we have to infer some of them.



C Phenotypic data analyses

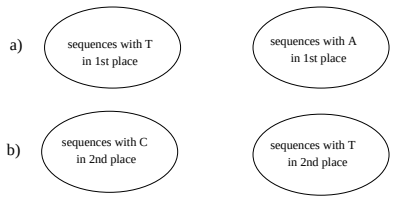
In this case our data consists of measurements from a number of individuals. For each individual we are given a DNA sequence and a characteristic trait, say. For example:

Indiv. A: Sequence ATCGA Size 1.9
Indiv. B: Sequence ATCGA Size 3.5
Indiv. C: Sequence ATCGA Size 3.1
Indiv. D: Sequence ATTGA Size 4.4
Indiv. E: Sequence ACCGA Size 1.2
Indiv. F: Sequence TTCGA Size 2.0

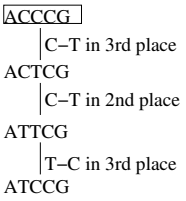
We see that the first 3 individuals have the same sequence. So, in our cladogram, they will all correspond to the same node, whose dataset has size 3.

Our aim is to draw some conclusion of the type "a mutation in the 3rd place of the DNA sequence under study is associated with a significant change in the size of individuals".

If the dataset is nicely behaved, then we would only have to try the 5 (or however many letters long the sequences are) possible clusterings:



etc., and find the clustering which best explains our data.



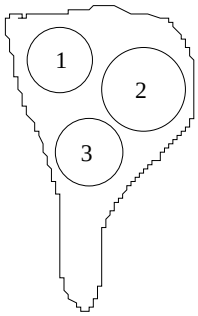
However, this isn't always possible. Consider the following history:

Here we have an example of homoplasy: 2 mutations happen in the same position (restriction site). If we try to split our data purely depending on the sequences (and not making inference about the underlying cladogram) that would lead to misleading conclusions. For that reason it is important that we try to infer the true underlying tree before trying to draw conclusions about associations.

The point of Nested Clade Analysis is to cluster the sequences and perform some ANOVAs in a sensible way which will be able to determine the significant mutation. The first level of nesting groups together all the youngest sequences (leaves) with their direct ancestor. At each nesting level, we join up one mutation up, until we have nested the whole tree.

D Phylogeographic data analyses

In this case our data looks like:
Indiv. A: Sequence ATCGA coordinates (1.2,4.5)
Indiv. B: Sequence ATCGA coordinates (3.2,1.4)
Indiv. C: Sequence ATCGA coordinates (3.6,1.6)
Indiv. D: Sequence ATTGA coordinates (4.5, 1.4)
Indiv. E: Sequence ACCGA coordinates (4.5, 1.8)
Indiv. F: Sequence TTCGA coordinates (2.5, 1.9)

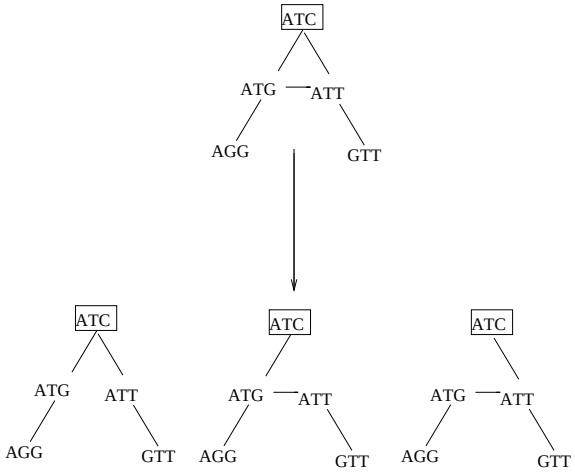


For such analyses it is very useful to use mitochondrial data. Mitochondrial DNA has no actual function. It lives inside the cell cytoplasm, and it is only inherited from one's mother. This means that it has some desirable properties, and also that we only have to look at 1 strand of DNA.

We want to cluster the data so that we have, say, 3 clusters as in the Figure on the right. If we conclude that, say, cluster 2 is the oldest, then we can say that clusters 1 and 3 are colonies of 2.

E Explanation of Figure 3.4.5

This graph is such that the true tree is assumed to be a subset of. E.g. the following graph would give us a choice of 3 possible true trees:



Appendix 4

Overview of MCMC methods

The idea behind Markov Chain Monte Carlo methods is to construct a Markov Chain whose stationary distribution is the posterior distribution of our parameters given our data. It is very useful in situations where the posterior distributions is not exact or is intractable, and it can deal with a large number of parameters which are all inferred simultaneously. The simplest case is when the size of the parameter space is fixed.

In order to obtain such an equilibrium distribution, we construct a Markov Chain such that:

1. Given that we are in a position $\boldsymbol{\theta}_t \in \Theta$ where $\boldsymbol{\theta}$ is the vector of parameters and Θ is the parameter space, we propose to jump to a value $\boldsymbol{\theta}_{t+1}$ according to a proposal distribution $q(\boldsymbol{\theta}, \boldsymbol{\theta'})$.

2. We accept this proposed move with probability $\min(1, A)$ where A is given below, otherwise set $\boldsymbol{\theta_{t+1}} = \boldsymbol{\theta_t}$

$$A = \frac{\mathbb{P}(data|\boldsymbol{\theta_{t+1}})p(\boldsymbol{\theta_{t+1}})q(\boldsymbol{\theta_{t+1}}, \boldsymbol{\theta_t})}{\mathbb{P}(data|\boldsymbol{\theta_t})p(\boldsymbol{\theta_t})q(\boldsymbol{\theta_t}\ , \boldsymbol{\theta_{t+1}})}$$

The chain described above (provided it is irreducible) is time-reversible (it can easily be checked that the detailed-balance equation holds) and its equilibrium distribution is equal to the posterior distribution of our parameters given the data.

If the number of parameters (in this case the number of significant mutations) is not fixed, have to use Reversible-Jump MCMC, where the size of the parameter space is allowed to vary from iteration to iteration. It is very useful when more than one model are proposed, with different parameter sizes.

In general, the acceptance ratio becomes:

$$A = \frac{\mathbb{P}(data|\boldsymbol{\theta_{t+1}})p(\boldsymbol{\theta_{t+1}})q(\boldsymbol{\theta_{t+1}}, \boldsymbol{\theta_t})}{\mathbb{P}(data|\boldsymbol{\theta_t})p(\boldsymbol{\theta_t})q(\boldsymbol{\theta_t}\ , \boldsymbol{\theta_{t+1}})} \left|J\right|,$$

where J is the Jacobian of the square map by which we assign new parameters when moving between spaces of different dimension.

In the case of clustering, this move is equivalent to proposing to increase or decrease the number of clusters by 1, and either combining 2 adjacent clusters or splitting an existing one, proposing new means using a Gibbs sampler.

This move is accepted with probability:

$$A = \frac{f(D|\boldsymbol{\mu'})p(\boldsymbol{\mu'_g})p(\boldsymbol{e'})q(\boldsymbol{\mu}'_g \to (\boldsymbol{\mu_i}, \boldsymbol{\mu_j}))q(\boldsymbol{e'} \to \boldsymbol{e})}{f(D|\boldsymbol{\mu})p(\boldsymbol{\mu_i})p(\boldsymbol{\mu_j})p(\boldsymbol{e})q((\boldsymbol{\mu_i}, \boldsymbol{\mu_j}) \to \boldsymbol{\mu}'_g)q(\boldsymbol{e} \to \boldsymbol{e'})}$$

since $|J|$, the Jacobian of the square map by which we assign new means, is 1. Here $q(\boldsymbol{\mu}'_g \to (\boldsymbol{\mu_i}, \boldsymbol{\mu_j}))$ is the proposal by which we generate new means (and similarly for the reverse proposal), $q(\boldsymbol{e'} \to \boldsymbol{e})$ is the proposal by which we decide to combine two clusters (or split one into 2).

Appendix 5

Wishart Distribution Generation

In order to generate a square matrix W of dimensions $d \times d$ which has a Wishart distribution with parameters (m,V), we first calculate the Choleski decomposition L of the matrix V, so that $LL^T = V$, where L^T is an upper-triangular matrix. The elements of L will be:

$$\begin{aligned} l_{ii} &= \sqrt{v_{ii} - \sum_{k=1}^{i-1} l_{ik}^2}, \\ l_{ji} &= \frac{v_{ji} - \sum_{k=1}^{i-1} l_{jk} l_{ik}}{l_{ii}}, \text{ if } j > i, \text{ and} \\ l_{ji} &= 0 \text{ if } j < i. \end{aligned}$$

It can be easily checked that a covariance matrix of the form

$$V = \begin{pmatrix} \sum_{i=1}^n x_{1i}^2 & \dots & \sum_{i=1}^n x_{1i} x_{di} \\ \vdots & \dots & \vdots \\ \sum_{i=1}^n x_{1i} x_{di} & \dots & \sum_{i=1}^n x_{di}^2 \end{pmatrix}$$

is always positive definite so the quantity under the root will be positive, so a Choleski decomposition exists. This is because $\forall \boldsymbol{a}$:

$$\begin{aligned} \boldsymbol{a}^T V \boldsymbol{a} &= a_1^2 \sum_{i=1}^n x_{1i}^2 + \ldots + a_d^2 \sum_{i=1}^n x_{di}^2 + a_1 a_2 \sum_{i=1}^n x_{1i} x_{2i} + \ldots + a_k a_l \sum_{i=1}^n x_{ki} x_{li} + \ldots \\ &= \sum_{i=1}^n \left(\sum_{j=1}^d a_j x_{ji} \right)^2 \geq 0 \end{aligned}$$

We then construct a matrix Z so that the elements z_{ii} across the diagonal are equal to $\sqrt{u}$, where $u \sim \chi^2(d, m - i + 1)$. We set the elements of Z in the lower left-hand triangle of the matrix to be equal to 0. Lastly, we set the elements in the top right-hand triangle of Z to be equal to w where $w \sim N(0, 1)$.

Then $W = (ZL^T)^T ZL^T$ will have a Wishart distribution with parameters (m,V).

In order to generate a matrix W' with distribution Inverse Wishart with parameters (m, V^{-1}), we simply generate a Wishart matrix W with parameters (m,V) and invert so that $W' = W^{-1}$.

Bibliography

G. Altekar, S. Dwarkadas, J. P. Huelsenbeck, and F. Ronquist. Parallel metropolis coupled markov chain monte carlo for bayesian phylogenetic inference. *Bioinformatics*, 20(4):407–415, 2004.

C.F. Aquadro, S.F. Desse, M.M. Bland, C.H. Langley, and C.C Andlaurie-Ahlberg. Molecular population genetics of the alcohol dehydrogenase gene region of drosophila melanogaster. *Genetics*, 114:1165–1190, 1986.

B. C. Emerson, S. Forgie, S. Goodacre, and P. Oromi. Testing phylogeographic predictions on an active volcanic island: Brachyderes rugatus (coleoptera: Curculionidae) on la palma (canary islands). *Molecular Ecology*, 15(2):449–458, 2005.

M. Hasegawa, H. Kishino, and T. Yano. Dating of the human-ape splitting by a molecular clock of mitchondrial dna. *J. Mol. Evol.*, 22:160–174, 1985.

B. H. Jordal, B. B. Normark, and B. D. Farrell. Evolutionary radiation of an inbreeding haplodiploid beetle lineage (curculionidae, scolytinae). *Biological Journal of the Linnean Society*, 71:483–499, 2000.

B. Larget and D. Simon. Markov chain monte carlo algorithms for the bayesian analysis of phylogenetic trees. *Molecular biology and evolution*, 16:750–759, 1999.

D. G. Lloyd and V. L. Calder. Multi-residue gaps, a class of molecular characters with exceptional reliability for phylogenetic analyses. *Journal of Evolutionary Biology*, 4(1):9–21, 1991.

B. Mau, M. Newton, and B. Larget. Bayesian phylogenetic inference via markov chain monte carlo methods. *Biometrics*, 55:1–12, 1999.

M.A. Newton, B. Mau B., and B. Larget. Markov chain monte carlo for the bayesian analysis of evolutionary trees from aligned molecular sequences. *Statistics in molecular biology and genetics*, 33:143–162, 1999.

S. Richardson and P. Green. On bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society. Series B (Methodological)*, 59(4):731–792, 1997.

M. A. Scataglini, A. A. Lanteri, and V. A. Confalonieri. Phylogeny of the pantomorus naupactus complex based on morphological and molecular data (coleoptera: Curculionidae). *Cladistics*, 2 (2):131, 2005.

A.S. Sequeira, A. A. Lanteri, M. A. Scataglini, V.A. Confalonieri, and B. D.Farrell. Are flightless galapaganus weevils older than the galapagos islands they inhabit? *Heredity*, 85:20–29, 2000.

M. Stephens. Dealing with label-switching in mixture models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 62(4):795–809, 2000.

S. Tavaré. Some probabilistic and statistical problems in the analysis of dna sequences. *Lectures on Mathematics in the Life Sciences,*, 17, 1986.

S. Tavaré. *Nature Encyclopedia of the Human Genome*. Nature Publishing Group, 2003.

A. R. Templeton. Nested clade analysis of phylogeographic data: testing hypotheses about gene flow and population history. *Molecular Ecology*, 7:381–397, 1998.

A. R. Templeton, E. Boerwinkle, and C. F. Sing. A cladistic analysis of phenotypic associations with haplotypes inferred from restriction endonuclease mapping. i. basic theory and an analysis of alcohol dehydrogenase activity in drosophila. *Genetics*, 117:343–351, 1987.

A. R. Templeton, K. A. Crandall, and C. F. Sing. A cladistic analysis of phenotypic associations with haplotypes inferred from restriction endonuclease mapping. iii. cladogram estimation. *Genetics*, 132(2):619–633, 1992.

A. R. Templeton and C. F. Sing. A cladistic analysis of phenotypic associations with haplotypes inferred from restriction endonuclease mapping. iv. nested analyses with cladogram uncertainty and recombination. *Genetics*, 134:659–669, 1993.