Statistical Methods for DNA Sequences: Detection of Recombination and Distance Estimation

Gráinne McGuire

Doctor of Philosophy
University of Edinburgh
1998

Acknowledgements

I would like to thank both my supervisors, Frank Wright and Mike Prentice, for all their help, ideas and encouragement over the last three years and for making this a very interesting project to work on. I would also like to acknowledge their assistance in preparing this manuscript.

I am grateful to all those at BioSS for providing a good atmosphere for studying. In particular, the colloquia and staff meetings were very beneficial, as was the encouragement to attend conferences and publicise my work.

Various people have discussed statistical, mathematical and computational aspects of this work over the last three years, and I would like to thank them. These include Nick Goldman, Chris Glasbey, Paul Sharp, Liz Bailes, Jon Bennett, Alan Bleasby, Peter Woollard, Steve Ferris and various anonymous referees. Many thanks also to my proof readers: Carol Smith and Esther Troy.

Abstract

Two problems in phylogenetics are considered here: the detection of evidence of recombination in DNA sequence multiple alignments and the improved estimation of confidence intervals for genetic distance estimators. Recombination between distinct species can result in mosaic sequences which often invalidate a simple tree-like model for between-species relationships. A graphical method based on pairwise distances and least squares is proposed as an initial scan of data sets for evidence of recombination prior to a phylogenetic analysis. A Bayesian model of recombination for data sets with a small number of species is described, which allows Hidden Markov model theory to be used to carry out computations (e.g., the calculation of the *maximum a posteriori* estimate).

Accurate estimation of confidence intervals for genetic distance estimators is important for comparing the relative rates of nucleotide substitution in different regions of DNA or for estimating the time since the most recent common ancestor. Two approximations to the sampling distributions of distance estimators are proposed. The first is a transformation of a normal density and may be applied to one-parameter models of nucleotide substitution only; this yields very accurate approximations to confidence intervals for a large range of distances. The second is the saddlepoint approximation which has a wider range of applicability (applicable to some two and three parameter models) and also performs well for a range of distances.

Table of Contents

Chapter 1

Introduction

1.1 Phylogenetic analysis using DNA sequence data – a brief introduction

Phylogenetics is concerned with finding relationships among species based on the degree of the similarity of their genetic information. It is a rapidly expanding field of research, since it is important in many biological applications to infer the relationships existing among species of plants and/or animals, or among strains of bacteria or viruses.

Genetic information is contained within nucleic acid, usually DNA. This is a linear molecule, consisting of a sequence of units called nucleotides, of which there are four types (A, C, G and T). A typical example of a subsequence of DNA might be $\ldots ACTTGAC \ldots$ Thus, DNA may be viewed as carrying the (encoded) instructions for life, written in an alphabet of four letters. This genetic information is sometimes contained in a single large DNA molecule or chromosome (e.g., in bacteria) or may be spread over several chromosomes (e.g., in humans). Chromosomes are typically several million nucleotides long, although most statistical analyses of DNA sequences involve subsequences consisting of a few thousand nucleotides or less.

Over time DNA sequences change through various types of mutations. These include the insertion or deletion of one or more nucleotides along a sequence or the substitution of one nucleotide for another. As an example of nucleotide substitution, suppose an original subsequence of DNA is $AGTC$. Following the substitution of a T for the G it becomes $ATTC$. Such events are examples of evolutionary change and may result in changes to the organism, be these detrimental or beneficial or neutral.

All species which are present today have arisen through a long period of evolution. The emergence of a new species may be postulated as resulting from the splitting of one species into two subspecies, which then independently accumulate evolutionary change. At some point, they have accumulated sufficient differences that they may be considered two distinct species. Therefore, all organisms alive today share ancestors in the past, and the relationships among species which are present today may be graphically

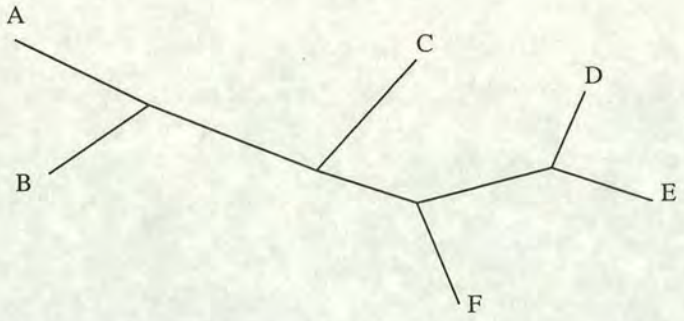


Figure 1.1: Example of a phylogenetic tree for six species.

described by a branching tree (the *phylogeny*). In principle, it should be possible to infer the phylogeny for a set of species from a comparison of their DNA sequences. Species with similar DNA sequences (e.g., as measured by the proportion of positions in the sequences with identical nucleotides) should be more closely related than species whose sequences differ by a greater degree. An example of a phylogenetic tree for six species, labelled A to F, is shown in Figure 1.1.

Clearly any methods for inferring a phylogeny (equivalently, the phylogenetic tree) should be statistical in nature since the evolutionary process is stochastic. The DNA sequences used in any analysis are subject to stochastic error, so that several trees, depicting different hierarchical relationships among the species, may be more or less equally good for a particular data set. This indicates that there is not enough information within the data set to give a more precise estimate of the relationships, and this must be acknowledged. Nonetheless, there are some biologists who oppose vehemently the use of statistics within phylogenetics, claiming that there is one true tree, and that only algorithms yielding point estimates (usually parsimony-based algorithms) are valid and find this true tree.

When inferring phylogenetic trees, only nucleotide substitution events are generally considered due to difficulties in modelling other evolutionary events. Three main classes of methods exist for inferring trees. The first, parsimony, considers the number of substitution events which must occur to result in a particular tree. The preferred tree is the one which requires the minimum number of changes. The second class of methods uses pairwise distances between the sequences rather than the raw sequence data. Therefore, the genetic distance between pairs of DNA sequences must first be estimated. This is based on the proportion of positions with different nucleotides in two DNA sequences. The formula for a genetic distance estimator may be derived from a model of the nucleotide substitution process. Once all the pairwise distances have been estimated, the phylogeny may then be inferred. Initially simple methods (e.g., cluster analysis) were used. However, with increasing computing power, more accurate, efficient and computationally more intensive methods were introduced. This, in turn,

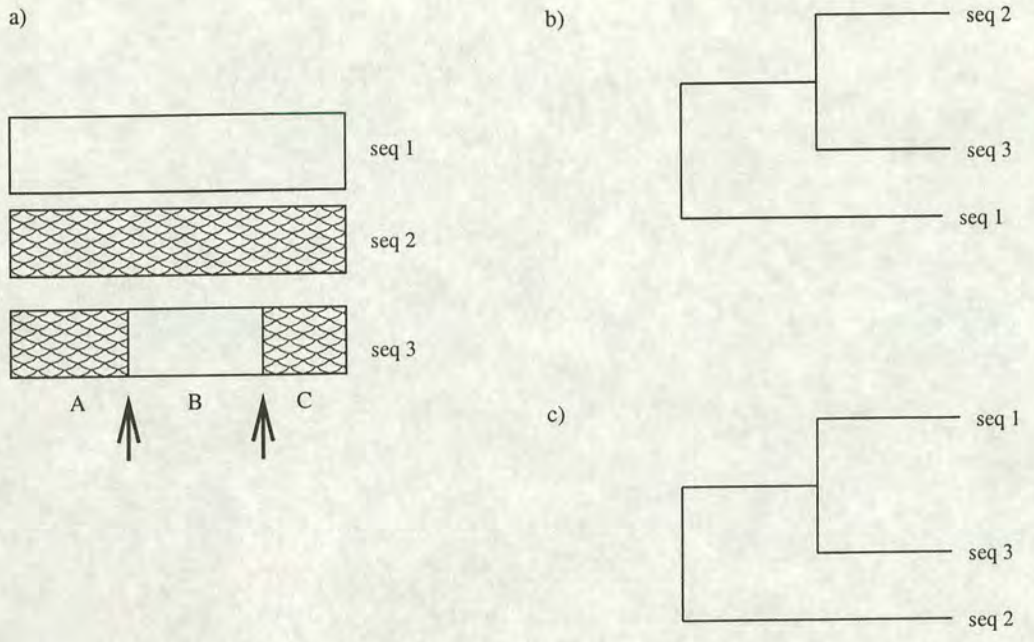


Figure 1.2: **a**: a simple example of recombination in a DNA sequence. At some point in the past, the central region of sequence 1 replaced that of sequence 2, forming the mosaic sequence 3. The arrows mark the limits of the recombination event, the *recombination breakpoints*. **b**: the relationships for parts A and C of the sequences. **c**: the relationships for part B of the sequences.

led to the practical application of the third class of procedures: maximum likelihood. This chooses the tree with the highest likelihood of producing the sequence data, given a particular model of nucleotide substitution.

1.2 Problems examined in this thesis

Two different problems are examined in this thesis. The first concerns the detection of evidence of recombination in DNA data sets. Recombination is the exchange of subsequences of DNA between different DNA sequences. To illustrate this, consider Figure 1.2. At some point in the past, the central subsequence of DNA in sequence 1 replaced that in sequence 2, forming sequence 3 (Figure 1.2a). A phylogenetic tree estimated from the two outer regions of the sequences would place sequences 2 and 3 together (Figure 1.2b), while a tree estimated using the central subsequence of DNA would have sequences 1 and 3 clustering together (Figure 1.2c). Using the entire sequence length to infer the tree would result in some sort of average between the two true relationships.

A similar effect is observed in general. Following a recombination event, the relationships within a data set often cannot be adequately described by a tree-like diagram. Indeed, recombination will often cause tree-estimation methods to give misleading results. Thus, it is important to detect recombination prior to a phylogenetic analysis so

that the DNA sequences can be split up into non-recombinant subsets and each subset analysed separately, allowing the true relationships to be inferred. Recombination is common in many bacteria (e.g., Listeria) and viruses (e.g., HIV) and has many important consequences. For example in AIDS research, it is important to know whether a strain of HIV is a distinct type, or a mosaic of two or more different types, as this has implications for vaccine design.

The second problem concerns inferences using estimators of genetic distance between pairs of DNA sequences. Genetic distance estimators are often derived from models for the nucleotide substitution process, such models usually being continuous-time, first-order Markov models with a state space consisting of the four nucleotides. These distance estimators depend on the proportion of observed differences between a pair of sequences. The simplest estimators depend only on the proportion of positions with non-identical nucleotides in the two sequences, while more complicated ones depend on the proportion of particular pairs of nucleotides observed. The observed numbers of different pairs of nucleotides in the sequences are observations from a multinomial distribution. To date, simple methods for estimating the variance of these estimators are used (e.g., the delta method is used to approximate the variance, based on the multinomial variance-covariance matrix) while normality is assumed to calculate confidence intervals. Improved methods for estimating the confidence intervals and sampling distribution of some of these estimators are considered here. This is important for applications such as estimating the time since two species last shared a common ancestor. This can sometimes be estimated from the distance between two species (if the rate of substitution is known). The confidence intervals for the distance estimator may be used to place confidence intervals on the time since the common ancestor, so improved accuracy of distance confidence intervals is important.

1.3 Plan of thesis

An introduction to DNA sequence data and phylogenetic trees is given in Chapter 2. The three main classes of estimating phylogenetic trees (parsimony, distance and maximum likelihood) are briefly discussed. Models for the nucleotide substitution process and the resulting distance estimators are also described. Finally an overview of some statistical tests is given.

Chapter 3 is also a review chapter, covering existing methods to detect evidence of recombination. The process of recombination is described and tests proposed in the literature are discussed. Their strengths and limitations are outlined.

A graphical method to detect evidence of recombination is presented in Chapter 4. This is based on pairwise distances and the least squares method of phylogenetic tree estimation. It is a procedure which may be used to quickly scan a data set for possible

recombination events prior to a phylogenetic analysis. It returns putative recombination breakpoints which may be tested using some of the methods described in Chapter 3. A simulation study was carried out and the method was applied to some real data sets to investigate the performance of this algorithm.

A more rigorous approach to the problem of detecting recombination is taken in Chapter 5. Here, Bayesian methodology and the theory of Hidden Markov models is used to find a mathematically tractable model of the location of recombination events within a DNA data set. For computational reasons, only data sets of four sequences are considered. A point estimate of the most probable phylogeny at each site is returned, thereby estimating both the location of recombination events and the effects on the branching pattern of the tree. The performance of this procedure was explored in a small simulation study.

The second problem, the improved estimation of confidence intervals for genetic distance estimators, is discussed in Chapter 6. Two approximations are suggested: one involves transforming normal probability quantiles, while the second uses the saddlepoint approximation to estimate tail probabilities. These approximations, where applicable, yield quite accurate confidence intervals over a wide range of distances and sequence lengths. They may also be used to approximate the sampling distribution of genetic distance estimators.

Suggestions for further work in these specific areas are given at the ends of Chapters 4 to 6. It is hoped that some of these could overcome current limitations to the suggested methodology.

Finally, Chapter 7 summarises the new procedures described in the previous three chapters and broadly looks at the possible direction of further research.

Chapter 2

Statistical Analysis of DNA Sequences

A brief overview of the statistical analysis of DNA sequence data is given in this chapter, concentrating on the inference of phylogenetic trees. This includes topics such as models of nucleotide substitution, estimation of genetic distance estimators and the three main classes of phylogenetic tree estimation: maximum parsimony, distance and maximum likelihood methods. For an excellent review of this area, Swofford et al. (1996) is recommended.

There are many applications of statistics in the analysis of DNA sequences beyond those mentioned above. For example, Markov models are used to analyse single sequences of DNA to search for over- or under-representation of particular short subsequences of nucleotides (Schbath et al., 1995). Another application is the alignment of sequences. This is briefly described in **2.3** since multiple alignments of DNA sequences are a prerequisite to the phylogenetic methods described below.

The chapter opens with a short description of phylogenetic trees and gives an idea of their biological uses. DNA sequence data is then discussed, this being the type of data for which the methodology described in this thesis is applicable. Algorithms and software to produce multiple alignments are briefly mentioned. Some of the procedures used to estimate trees are described. Particular emphasis is placed on models of nucleotide substitution, distance and maximum likelihood methods of tree inference, since these methods will be used later. Finally, some statistical tests are briefly discussed.

2.1 Phylogenetic trees

Phylogenetic methods infer the hierarchical relationships existing among a set of species, or strains of bacteria or viruses. These relationships may be represented by a phylogenetic tree.

Phylogenetic trees have a wide range of applications for biologists. For example, they may be used to estimate the ancestry of the human race (they can provide infor-

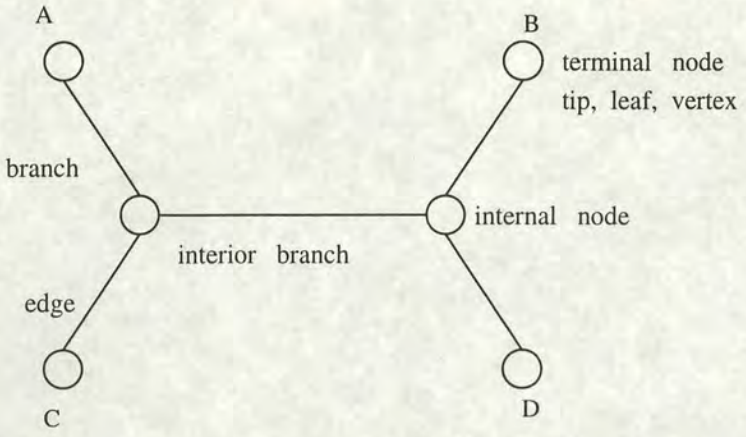


Figure 2.1: Some of the terms describing the components of a phylogenetic tree.

mation on the 'Out of Africa' hypothesis for instance) and of other species. Another use is in the tracing of the course of epidemics. For instance, a recent case in Florida involved a dentist who was HIV positive and was accused of passing on the virus to his patients. The evidence was assessed using phylogenetic trees (Hillis et al., 1996).

Most phylogenetic methods result in the inference of an unrooted tree (a phylogeny in which the earliest point in time is not identified). The components of a tree are known by various names, including the mathematical terms from graph theory. For example, the contemporary *taxa* (the species or sequences in the data set) correspond to terminal or external nodes. These may also be referred to as leaves, tips or vertices. Branching points within a tree (representing ancestral sequences) are called internal nodes, and sometimes vertices. The edges of the graph (the lines connecting nodes in the tree) are often known as branches. A distinction is sometimes drawn between the branches incident to a terminal node, and those connecting internal nodes only. The latter are referred to as interior branches. A phylogenetic tree for four taxa (A, B, C, D) with labels for some of these components is shown in Figure 2.1.

If only three branches are incident to an internal node, then this is said to be a bifurcation, or a dichotomy. If more than three branches are connected to a node, then this is a multifurcation (polytomy). A tree with bifurcations at all internal nodes may be called binary, fully resolved or strictly bifurcating. A special case of a multifurcating node is a *star* tree or phylogeny; this contains only one internal node, with branches radiating out from it to each of the tips.

Sometimes, only the branching order in a tree is of interest. This is often referred to as the *topology* and excludes information on the branch lengths. When counting the number of possible trees for T taxa (tips), it is really the number of possible topologies (branching patterns) that is being counted. Most phylogenetic tree estimation methods infer unrooted binary trees; the problem of counting all such possible topologies was considered by Edwards and Cavalli-Sforza (1964) and Felsenstein (1978a). An unrooted

bifurcating tree with T terminal nodes has $T-2$ internal nodes and $2T-3$ branches in total. Of these, $T-3$ are interior. To count the number of possible trees, consider the following recursion. For two species, there is only one possible branch on which to add a new taxa (tip). Thus there is only one possible three-taxa tree. Consider now a tree containing $k-1$ taxa. There are $2k-5$ branches to which the k^{th} species could be added ($k-1$ branches leading to a tip and $k-4$ interior branches). Hence, the total number of distinct strictly bifurcating trees for T taxa is given by

$$N(T) = \prod_{i=3}^{T}(2i-5). \tag{2.1}$$

This relationship may also be used to count the number of rooted trees. Placing a root on an unrooted tree adds one more internal node, and one more interior branch. Since the root may be placed along any of the $2T-3$ branches, the number of possible rooted trees is increased by a factor of $2T-3$.

As indicated above, the vast majority of phylogenetic tree estimation methods yield unrooted trees. However, it is possible to root trees using a technique called *outgroup rooting*. This involves including one or more sequences in the analysis which are known to be an outgroup to the original data set (i.e., are relatively distantly related to the taxa in the data set). The location at which the outgroup joins the unrooted tree implies a root for the original data. It is important to note that by choosing the outgroup, the assumption is made that the remaining taxa are monophyletic (all descending from a common ancestor). If this is invalid, the tree will be incorrectly rooted.

2.2 DNA sequence data

Various types of molecular data may be used to infer phylogenetic trees. Possibilities include restriction endonuclease data and allozyme data; for more details, see Swofford et al. (1996) and references therein. Since the advent of polymerase chain reaction (PCR), the amount of available DNA sequence data has rapidly increased, and has become widely and freely available in databases such as GenBank and EMBL (these contain approximately 500,000 entries, each, on average, 1000 nucleotides long). The procedures in this thesis were developed with DNA sequence data in mind; thus a description of this data is required.

Apart from RNA viruses, the hereditary information of all living organisms is carried by *deoxyribonucleic acid* (DNA) molecules. These usually consist of two complementary chains twisted around each other to form a right-handed helix. Each chain is a linear sequence consisting of four nucleotides or bases. These may be divided into two groups, based on their biochemical properties:

the **purines**: adenine (A) and guanine (G)

Table 2.1: The universal genetic code

codon	amino acid	codon	amino acid	codon	amino acid	codon	amino acid
TTT	Phe (F)[a]	TCT	Ser (S)	TAT	Tyr (Y)	TGT	Cys (C)
TTC	Phe (F)	TCC	Ser (S)	TAC	Tyr (Y)	TGC	Cys (C)
TTA	Leu (L)	TCA	Ser (S)	TAA	Stop[b]	TGA	Stop
TTG	Leu (L)	UCG	Ser (S)	UAG	Stop	UGG	Trp (W)
CTT	Leu (L)	CCT	Pro (P)	CAT	His (H)	CGT	Arg (R)
CTC	Leu (L)	CCC	Pro (P)	CAC	His (H)	CGC	Arg (R)
CTA	Leu (L)	CCA	Pro (P)	CAA	Gln (Q)	CGA	Arg (R)
CTG	Leu (L)	CCG	Pro (P)	CAG	Gln (Q)	CGG	Arg (R)
ATT	Ile (I)	ACT	Thr (T)	AAT	Asn (N)	AGT	Ser (S)
ATC	Ile (I)	ACC	Thr (T)	AAC	Asn (N)	AGC	Ser (S)
ATA	Ile (I)	ACA	Thr (T)	AAA	Lys (K)	AGA	Arg (R)
ATG	Met (M)	ACG	Thr (T)	AAG	Lys (K)	AGG	Arg (R)
GTT	Val (V)	GCT	Ala (A)	GAT	Asp (D)	GGT	Gly (G)
GTC	Val (V)	GCC	Ala (A)	GAC	Asp (D)	GGC	Gly (G)
GTA	Val (V)	GCA	Ala (A)	GAA	Glu (E)	GGA	Gly (G)
GTG	Val (V)	GCG	Ala (A)	GAG	Glu (E)	GGG	Gly (G)

[a]Amino acids are denoted by their standard three-letter and one letter abbreviations

[b]Stop codons cause the transcription process from DNA sequences to amino acids to stop. Thus, they mark the end of a protein coding region

the **pyrimidines**: cytosine (C) and thymine (T).

DNA may be written as a linear string of these nucleotides, e.g., $ACTTGA\ldots$ Such sequences are often said to be x base pairs (bp for short) long, where x is the number of nucleotides in the sequence.

RNA (ribonucleic acid) exists as both a double- and a single-stranded molecule. It is similar to DNA, but uses the nucleotide *uracil* (U) instead of thymine (T). There are several types of RNA molecule. One type (mRNA) is involved in protein production.

Some subsequences of DNA correspond to genes or parts of genes which carry instructions for making proteins. In the protein-coding region, the DNA is arranged in triplets, called codons, with $4^3 = 64$ possible arrangements. Each codon corresponds to a particular amino acid (the building blocks of proteins). In nearly all species, the correspondence follows a universal code (see Table 2.1). Note that this is a degenerate code: most of the 20 amino acids are encoded by more than one codon.

Some genes are interrupted by non-coding regions of DNA, which are known as introns. See Figure 2.2 for a simple example of two genes containing introns, as well as a non-protein coding sequence of DNA separating the two genes (an intergenic region). Introns are ignored in the process of reading protein coding information from the DNA

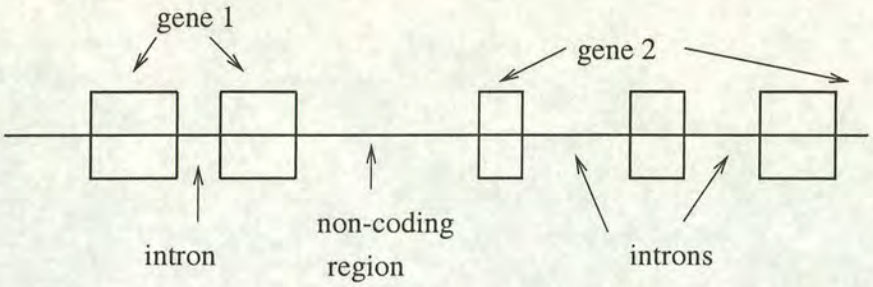


Figure 2.2: schematic diagram of a DNA sequence. The boxes correspond to the protein-coding sequences of two genes. Within each gene are non-coding regions called introns, while the genes are separated by a non-coding sequence (an intergenic region).

template and lack the triplet structure of the protein-coding genes, as do intergenic regions. More information on DNA and its structure may be found in Li and Graur (1991, Chapter 1).

All organisms must copy their DNA in order to reproduce. The replication mechanism is generally accurate but occasionally, a mutation occurs. This could be due to the substitution of one nucleotide for another, or insertion or deletion events involving one or more nucleotides. Some of these changes will be deleterious, and the organism may fail to reproduce, with the result that the mutation is not passed onto the next generation (i.e., the mutation is removed by natural selection). On the other hand, some of these mutations may not affect the organism greatly, or may even be beneficial, and thus, the organism will pass on its genetic material (including the mutation) to the next generation. Mutations occurring in the DNA of a mating population will add to the variability of the population. If, however, the population splits into two distinct subpopulations, each group will accumulate mutations independently of each other. Eventually a sufficient amount of change may occur to make the two subpopulations so different from each other that they are unable to interbreed. At this point they have become two different species. If one individual is sampled from each species, modelling evolution as a series of bifurcation events is justified.

In order to model evolution well, all possible mutation processes should be included in any model. Unfortunately, it is difficult to model all but the nucleotide substitution events. Consequently, the data used in phylogenetic analysis are generally those parts of a DNA sequence which are believed to have arisen by nucleotide substitution. Subsequences that appear to have been subject to other evolutionary processes are excluded. Thus, models of nucleotide substitution only are, in general, applied to the data.

Referring back to the degeneracy of the codon-amino acid code in Table 2.1, it is seen that changes in the third position of the codon are not as likely to cause a change in the amino acid encoded (conversely a change in the second position almost always causes a change in the resulting amino acid). Hence, nucleotide substitution events in

```
raw sequences ->   alignment      (a)
   ATCGTCAG   ->  ATCG--TCAG
                  | ||  || |
   AGCGTATCG  ->  AGCGTATC-G
```

$$\text{score} = \text{no. of matches} - \text{no. of gaps} \times \text{gap presence penalty} - \text{no. of gap spaces} \times \text{gap length penalty} \quad \text{(b)}$$

$$\begin{aligned} \text{score} &= 6 - 2(3) - 3(0.5) \\ &= -1.5 \end{aligned} \quad \text{(c)}$$

Figure 2.3: example of a pairwise alignment problem employing user-provided gap presence and gap length penalties.

the third codon position are less likely to be removed by natural selection. This means that a greater rate of substitution is generally observed in the third codon position.

Higher rates of nucleotide substitution are generally observed in non-coding regions of DNA, either between genes (intergenic regions) or within genes (the introns). Since these regions do not carry instructions for protein products, changes within them are unlikely to be deleterious to the organism, and are often allowed to persist. It is important in a good phylogenetic analysis to take account of the organisation of the DNA being analysed, and allow for possible heterogeneous rates of substitution.

As well as DNA sequences, protein sequences are often used in phylogenetic analysis. These may be written as a linear string of letters, these being the twenty letters corresponding to the twenty amino acids. Statistical methods for DNA sequences have been applied to protein sequences, but there is a significant increase in the computational burden due to the increased number of characters.

2.3 Multiple alignments

Ideally, the alignment of DNA or amino acid sequences and the estimation of the hierarchical relationships existing between them should be carried out simultaneously, since the accurate estimation of among-species relationships is important to yield a good multiple alignment, while a sensible multiple alignment is required as input to a phylogenetic tree estimation method. Otherwise the old maxim: 'garbage in, garbage out' applies. Unfortunately, this would be a very difficult computational exercise. Therefore, the two steps are carried out separately.

To illustrate an approach to the alignment problem, consider the two sequences shown in Figure 2.3a. The aim is to match the two sequences as closely as possible. In order to do this, it may be necessary to introduce gaps to allow for the insertion

and deletion of nucleotides. To prevent too many gaps being introduced, penalties for introducing a gap and also for the length of a gap are assigned. The aim then is to maximise a score function such as that in Figure 2.3b. Typically, each identical pair of nucleotides is assigned a score of one, while mismatched pairs score zero. Gap penalties might be three, with a penalty of 0.5 for each position or site in a sequence in a gap. Thus, in the alignment in Figure 2.3a, there are two gaps, each incurring a penalty of three, while three sites lie in gaps (penalty of 0.5). This leads to the score shown in Figure 2.3.

It is straightforward to find the optimal alignment for a pair of sequences. A similar approach may be used to find the best alignment for three or more sequences but this problem is much harder. A number of computer programs exist which implement different approaches to the problem of aligning a set of DNA or amino acid sequences. Given a set of protein coding DNA sequences, it is generally better to input the corresponding amino acid sequences into the multiple alignment program and align these. The result can easily be translated back to nucleotides. For non-protein coding sequences, this is not possible of course, so the DNA sequences must be aligned. For a review of some of these programs, see McClure et al. (1994).

ClustalW (and its earlier versions), in particular, is widely used, and has been used later on in this thesis to align sets of DNA sequences. The algorithm used in this program has been described by Thompson et al. (1994), and consists of three main steps:

1. All possible pairs of sequences are aligned separately and a measure of divergence for each pair is calculated, resulting in a pairwise distance matrix;

2. A guide tree is calculated from this distance matrix using a clustering method (Neighbor Joining, see **2.7.5**);

3. The sequences are progressively aligned according to the guide tree, with the most closely related species being aligned first.

Finding a good multiple alignment is very important but can be a time-consuming task. As a rule of thumb, in any good phylogenetic analysis an equal amount of time should be spent on the multiple alignment as on the phylogenetic tree. For the work described in this thesis, it is assumed that the multiple alignment is known beforehand. This is automatically true for the simulated data. This assumption should not be a problem for the real data sets used to illustrate points in the later chapters either, since these consist of closely-related sequences which were easily aligned.

Methods for constructing phylogenetic trees are now considered. Three classes are discussed: maximum parsimony, maximum likelihood and distance methods. There are

a few procedures which fall outside of these classes, but these are not frequently used and will not be described here.

2.4 Parsimony methods for constructing phylogenetic trees

Phylogenetic tree reconstruction methods which employ the use of the principle of parsimony have been the most widely used by biologists to date. A parsimony optimality criterion is defined and the best trees are those which minimise this criterion, and are known as the most parsimonious trees.

In line with most tree estimation procedures, the maximum parsimony method assumes that each site (sequence position or column in the multiple alignment) evolves independently of the others. This allows the value of the parsimony optimality criterion to be found at each position in the multiple alignment, and these values may be summed over the entire data set. In general, parsimony methods select those trees which minimise the total tree length (the number of substitutions required to explain a given set of data). In mathematical terms, the solution to the parsimony problem is the set of all trees τ, such that the following is minimised:

$$L(\tau) = \sum_{k=1}^{B} \sum_{j=1}^{N} \text{diff}(x_{k_1 j}, x_{k_2 j}) \tag{2.2}$$

where $L(\tau)$ is the length of tree τ;

B is the number of branches;

N is the total sequence length;

k_1, k_2 are the two nodes incident to each branch k;

$x_{k_i j}$, $(i = 1, 2)$ represent either elements of the input data matrix or optimal character-state assignments made to internal nodes;

$\text{diff}(y, z)$ is a function specifying the cost of a transformation from state y to state z along any branch.

(Swofford et al., 1996).

There are many forms of the parsimony criterion in use. For details, see Swofford et al. (1996) and Felsenstein (1988) and references therein. Supporters of parsimony often claim that the use of this procedure requires no substantive assumptions about the evolutionary process, an assertion which is certainly questionable. While no explicit model of evolution is assumed, parsimony implicitly assumes that evolutionary change is very rare. Thus, multiple changes at a site, which would mislead the algorithm, are assumed to be very unlikely to occur. If the assumptions of parsimony are met, the method will perform well, and may be viewed as an approximation to the maximum likelihood method (Edwards, 1996). If the assumptions are not met, then parsimony will often be inconsistent (i.e., the estimate converges on an incorrect tree as more and

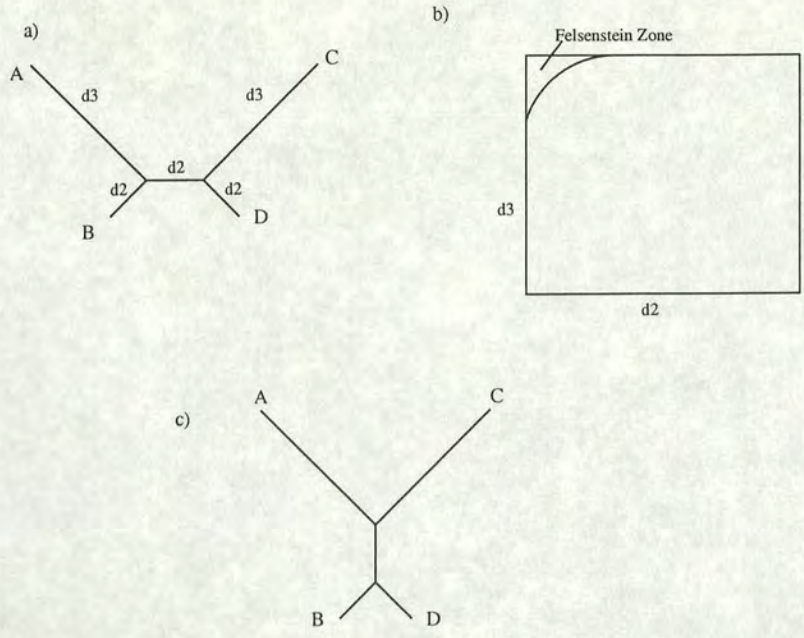


Figure 2.4: **a**: four species tree with two distinct branch lengths, d_2 and d_3, as shown. **b**: the Felsenstein zone for d_2 and d_3 (where parsimony methods will consistently give the wrong answer for a tree such as that in (a). **c**: the incorrect tree estimated when the branch lengths lie in the Felsenstein zone.

more data are used). This problem was highlighted by Felsenstein (1978b) for a four-taxa tree lying in what has since been termed the Felsenstein zone. An example of this is shown in Figure 2.4. With increasing sequence length, parsimony will be more likely to estimate the incorrect tree shown in Figure 2.4c.

2.5 Models of the nucleotide substitution process

The nucleotide substitution process is generally modelled by first-order, stationary, continuous-time Markov models. Below, a brief outline of continuous-time Markov models is presented, before the specific models used to depict the nucleotide substitution process are described.

2.5.1 Continuous-time, first-order Markov chains

An excellent introduction to continuous-time, first-order Markov chains is given in Grimmett and Stirzaker (1992, Chapter 6). Some of the basic theory is presented here; this is used later in the formulation of models for the nucleotide substitution process.

Let $X = \{X(t) : t \in [0, \infty)\}$ be a family of random variables which take values in some countable state space, S. Then X is a continuous-time, first-order Markov chain

if it satisfies

$$\begin{aligned}\text{Prob}\big(X(t_n) = j | X(t_1) = i_1, \ldots, X(t_{n-1}) = i_{n-1}\big) \\ = \text{Prob}\big(X(t_n) = j | X(t_{n-1}) = i_{n-1}\big)\end{aligned}$$

for all $j, i_1, \ldots, i_{n-1} \in S$ and any sequence $t_1 < t_2 < \ldots < t_n$ of times.

Many features of a continuous-time Markov chain are quite similar to a discrete-time chain. For example, the transition probability, $P_{ij}(s,t)$ is defined as

$$P_{ij}(s,t) = \text{Prob}\big(X(t) = j | X(s) = i\big)$$

for $s \leq t$. The chain is said to be *homogeneous* if $P_{ij}(s,t) = P_{ij}(0, t-s)$. for all i, j, s, t. In this case, $P_{ij}(s,t)$ may be more conveniently written as $P_{ij}(t-s)$.

To describe a homogeneous discrete-time Markov chain, it is necessary to specify the one-step transition probability matrix (i.e., that matrix containing the entries $P_{ij}(1)$ for all i, j). For a continuous-time Markov chain, there is no obvious unit of time, so instead a matrix, $\mathbf{R}$, giving the instantaneous rates of change is used. For the Markov chains considered below, this rate matrix has the property that $\mathbf{R}\mathbf{1}^T = \mathbf{0}^T$ where $\mathbf{1}$ and $\mathbf{0}$ are row vectors consisting of ones and zeros respectively. Alternatively, this condition may be written as $\sum_j r_{ij} = 0$, where r_{ij} are the entries of the matrix $\mathbf{R}$.

It can be shown, using the Kolmogorov forward equations, that $\mathbf{P}'_t = \mathbf{P}_t\mathbf{R}$ ($\mathbf{P}_t$ being the matrix with entries $P_{ij}(t)$ while $\mathbf{P}'_t$ is the matrix $d\mathbf{P}/dt$ and has entries $P'_{ij}(t)$). Similarly, the backward equations yield that $\mathbf{P}'_t = \mathbf{R}\mathbf{P}_t$. Subject to the condition that $\mathbf{P}_0 = \mathbf{I}$, these equations are solved by

$$\mathbf{P}_t = \sum_{n=0}^{\infty} \frac{t^n}{n!}\mathbf{R}^n$$

where $\mathbf{R}^0 = \mathbf{I}$. This is usually written as $\mathbf{P}_t = \exp(\mathbf{R}t)$ and provides a easy way of obtaining the transition probabilities for any given time of length t.

A vector $\boldsymbol{\pi}$ is a stationary distribution of a continuous-time Markov chain if $\pi_j \geq 0$, $\sum_j \pi_j = 1$ and $\boldsymbol{\pi} = \boldsymbol{\pi}\boldsymbol{P}_t$ for all $t \geq 0$. This condition is satisfied if and only if $\boldsymbol{\pi}\boldsymbol{R} = \mathbf{0}$. The latter allows the simple calculation of the stationary distribution for a Markov chain.

2.5.2 Continuous-time Markov models for the nucleotide substitution process

As mentioned above, the nucleotide substitution process is often modelled by a first-order, continuous-time Markov chain, where the chain takes values in the finite state space $S = \{A, C, G, T\}$. In addition to the basic properties, described above, of such chains, the following simplifying assumptions about the substitution process are usually made (Rodriguez et al., 1990; Kelly, 1994):

1. *Sites in the sequence are identically distributed.* Most models assume that the rates of nucleotide substitution at all sites are equal. Thus, the same rate matrix applies to all sites in the sequence;

2. *Sites evolve independently of each other*;

3. *The nucleotide substitution process is reversible* (i.e., $\pi_i p_{ij}(t) = \pi_j p_{ji}(t)$ where π_i, $i = A, C, G, T$ is the stationary probability nucleotide i). This results in constraints on the form of the transition probability matrix, thereby reducing the number of parameters. It explains why likelihood and distance methods estimate unrooted trees, since reversible models do not specify the direction of time;

4. *The nucleotide substitution process is at equilibrium.* This means that the frequencies of the nucleotides in the sequence correspond to the stationary distribution of the nucleotides.

Hence, the rate matrix for a general, time-reversible model of this type is given by

$$\mathbf{R}_{GTR} = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{cccc} A & C & G & T \\ - & a\pi_C & b\pi_G & c\pi_T \\ a\pi_A & - & d\pi_G & e\pi_T \\ b\pi_A & d\pi_C & - & f\pi_T \\ c\pi_A & e\pi_C & f\pi_G & - \end{array} \tag{2.3}$$

where the diagonal elements are given by

$$r_{ii} = \sum_{j \neq i} -r_{ij} \tag{2.4}$$

since the rows of a rate matrix must sum to zero. π_i, $i = A, C, G, T$ is the stationary nucleotide frequency of nucleotide i and $a, b, \ldots, f$ are the rate parameters, specifying the relative rates of change between two nucleotides. Note that this is a nine parameter model, the parameters being the six rate parameters and three nucleotide frequencies (the four nucleotide frequencies must sum to one; this constraint reduces the number of parameters by one). This model has been discussed by Lanave et al. (1984), Rodriguez et al. (1990) and Li and Gu (1996).

Many simpler cases of this model exist and some of these are examined below, starting with the simplest versions and proceeding upwards towards this nine parameter model. In many cases, these steps represent the historical order in which the models were proposed.

Jukes and Cantor (1969) were the first to suggest the Markov framework for modelling sequence evolution. They proposed a very simple model, with the stationary nucleotide frequencies all being equal ($\pi_i = 0.25$, $i = A, C, G, T$). They also assumed that all changes were equally likely ($a = b = \ldots = f$). Thus, the instantaneous rate

matrix in (2.3) reduces to the simple form

$$\mathbf{R}_{JC} = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{cccc} A & C & G & T \\ - & \alpha & \alpha & \alpha \\ \alpha & - & \alpha & \alpha \\ \alpha & \alpha & - & \alpha \\ \alpha & \alpha & \alpha & - \end{array} \tag{2.5}$$

where the diagonal elements are again calculated using (2.4).

This model is, of course, an oversimplification, as it is well known that nucleotide substitutions generally do not all occur at the same rates. In particular, changes within either the purine or pyrimidine nucleotide classes (*transitions*) tend to occur more frequently than *transversions* (substitutions between classes). This was recognised by Kimura (1980) when he proposed his two-parameter model. Like the Jukes-Cantor model, he assumed that the equilibrium frequencies of the nucleotides were all equal, but he allowed transitions ($A \leftrightarrow G$, $C \leftrightarrow T$) and transversions ($A \leftrightarrow C$, $A \leftrightarrow T$, $C \leftrightarrow G$, $G \leftrightarrow T$) to occur at different rates. In (2.3) this is equivalent to setting $b = e$ and $a = c = d = f$ and yields a rate matrix of the following form:

$$\mathbf{R}_{K2P} = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{cccc} A & C & G & T \\ - & \beta & \alpha & \beta \\ \beta & - & \beta & \alpha \\ \alpha & \beta & - & \beta \\ \beta & \alpha & \beta & - \end{array} \tag{2.6}$$

In the literature, this is generally referred to as the Kimura two Parameter model.

Meanwhile, Felsenstein (1981) extended the Jukes-Cantor model in another way. He supposed that all changes still occurred at the same rate, but allowed the nucleotide frequencies to be unequal. This model, known as the Felsenstein 81 model, has the following rate matrix

$$\mathbf{R}_{F81} = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{cccc} A & C & G & T \\ - & \gamma\pi_C & \gamma\pi_G & \gamma\pi_T \\ \gamma\pi_A & - & \gamma\pi_G & \gamma\pi_T \\ \gamma\pi_A & \gamma\pi_C & - & \gamma\pi_T \\ \gamma\pi_A & \gamma\pi_C & \gamma\pi_G & - \end{array} \tag{2.7}$$

obtained from (2.3) by setting $a = b = \ldots = f = \gamma$.

The next logical step was to combine the extensions in both the Kimura two Parameter and the Felsenstein 81 models to produce a two parameter model with unequal nucleotide frequencies. This was done in two ways; Hasegawa et al. (1985) proposed a rate matrix of the form

$$\mathbf{R}_{HKY85} = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{cccc} A & C & G & T \\ - & \upsilon\pi_C & s\pi_G & \upsilon\pi_T \\ \upsilon\pi_A & - & \upsilon\pi_G & s\pi_T \\ s\pi_A & \upsilon\pi_C & - & \upsilon\pi_T \\ \upsilon\pi_A & s\pi_C & \upsilon\pi_G & - \end{array} \tag{2.8}$$

where s represents the rate of transitions and v the rate of transversions. This model is often referred to as the HKY85 model.

This model is not very tractable mathematically; for example it is impossible to find a closed form solution for the genetic distance (see **2.7.1**). Felsenstein suggested another form for use in his DNAML program from the PHYLIP package (Felsenstein, 1993). This model is known as the Felsenstein 84 model, and was described by Felsenstein and Churchill (1996). It is computationally much simpler than the HKY85 model.

There are two types of nucleotide substitution event in the Felsenstein 84 model:

Type I either no change, or a transition (essentially a nucleotide is drawn at random from within the purine $[A, G]$ or the pyrimidine $[C, T]$ class to replace the current nucleotide, the choice of class being that of the current nucleotide);

Type II no change, a transition or a transversion (a nucleotide is drawn at random from the set of all nucleotides to replace the current one).

If the type I event occurs at a rate ρ, while the type II event occurs at a rate γ then the instantaneous rate matrix may be written as:

$$\mathbf{R}_{F84} = \begin{array}{c|cccc} & A & C & G & T \\ A & - & \gamma\pi_C & \frac{\rho\pi_G}{\pi_A+\pi_G} + \gamma\pi_G & \gamma\pi_T \\ C & \gamma\pi_A & - & \gamma\pi_G & \frac{\rho\pi_T}{\pi_C+\pi_T} + \gamma\pi_T \\ G & \frac{\rho\pi_A}{\pi_A+\pi_G} + \gamma\pi_A & \gamma\pi_C & - & \gamma\pi_T \\ T & \gamma\pi_A & \frac{\rho\pi_C}{\pi_C+\pi_T} + \gamma\pi_C & \gamma\pi_G & - \end{array} \tag{2.9}$$

As before the diagonal elements of $\mathbf{R}_{F84}$, r_{ii}, are given by (2.4).

The transition-transversion ratio is an important quantity in these two parameter models, and specifies the relative rates of transitions and transversions. For the Kimura two Parameter model, this ratio has a very simple form:

$$ts/tv = \frac{\alpha}{2\beta}. \tag{2.10}$$

The expression is more complicated for the Felsenstein 84 model, as it depends on functions of the nucleotide frequencies:

$$ts/tv = \frac{\rho A + \gamma B}{\gamma C} \tag{2.11}$$

where

$$\begin{aligned} A &= \frac{\pi_A\pi_G}{\pi_A+\pi_G} + \frac{\pi_C\pi_T}{\pi_C+\pi_T} \\ B &= \pi_A\pi_G + \pi_C\pi_T \\ C &= (\pi_A+\pi_G)(\pi_C+\pi_T). \end{aligned} \tag{2.12}$$

Note that if $ts/tv = 0.5$ for the Kimura two Parameter model, then the Jukes-Cantor model is obtained, while if $ts/tv = B/C$ for the Felsenstein 84 model, this model

simplifies to the Felsenstein 81 case. The transition-transversion ratio for the HKY85 model is given by sB/vC, with B and C being defined as in (2.12).

Various other special cases of (2.3), the general time-reversible model, have been proposed. Kimura (1981) proposed a three parameter model. Again the nucleotide frequencies were equal, but there were three rate parameters: one rate for transitions and two rates for transversions. Tamura and Nei (1993) developed a different three parameter model; this had one parameter for transversions, but two rate parameters for transitions [equivalent to letting $b = c = d = e$ in the rate matrix, $\mathbf{R}_{GTR}$, given in (2.3)]. The nucleotide frequencies were allowed to be unequal. Zharkikh (1994) described a model with six rate parameters, but with the equilibrium frequencies of the nucleotides all equal.

Once the rate matrix for a particular model has been specified, the transition probability matrix, $\mathbf{P}_t$, may easily be found using the relationship

$$\mathbf{P}_t = e^{\mathbf{R}t}. \tag{2.13}$$

Thus expressions for the transition probabilities may be easily found by hand, or by using a symbolic algebra package such as MAPLE (MAPLE V release 4, Waterloo Maple Software, Waterloo). As an example of this, consider the transition probabilities for the Kimura two Parameter model. Recalling that $P_{ij}(t)$ is the probability that a particular site initially with nucleotide i has nucleotide j after a time t, then from (2.13) it emerges that

$$P_{ij}(t) = \begin{cases} \frac{1}{4} + \frac{1}{4}e^{-4\beta t} + \frac{1}{2}e^{-2(\alpha+\beta)t} & \text{if } i = j, \\ \frac{1}{4} + \frac{1}{4}e^{-4\beta t} - \frac{1}{2}e^{-2(\alpha+\beta)t} & \text{if } i \neq j, \text{ transition}, \\ \frac{1}{4} - \frac{1}{4}e^{-4\beta t} & \text{if } i \neq j, \text{ transversion}. \end{cases}$$

2.6 Maximum likelihood methods for estimating phylogenetic trees

Maximum likelihood was proposed for use in phylogenetic inference by Cavalli-Sforza and Edwards (1967) and was first used for nucleotide sequences by Felsenstein (1981). Initially, a major drawback to the method was the computational burden it imposed. While this is still a problem for larger data sets, with increasing computer power maximum likelihood is becoming more widespread in use.

Maximum likelihood has some attractive properties. It is consistent, efficient and often robust to violation of assumptions. It also generates estimates with lower sampling variances even with short sequences. Most models of nucleotide substitution make the assumption that the substitution processes at each site are the same; while this is unlikely to be exactly true, it is reasonable that the processes at each site will have much in common and thus the evolution of sequences can be described by just a few

parameters. Consequently, tree inference using maximum likelihood tends to outperform parsimony and distance methods (Kuhner and Felsenstein, 1994; Huelsenbeck, 1995). Simulation studies have also found maximum likelihood to be quite robust. For example, Schöniger and von Haeseler (1995) found that violations of the assumption of independence between sites did not affect the performance of maximum likelihood to a great extent.

To illustrate how to calculate the likelihood for a particular phylogenetic tree, consider the four sequence alignment in Figure 2.5a and the tree shown in Figure 2.5b. The assumption that sites evolve independently of each other simplifies the calculation of the likelihood: the likelihoods at each site may be calculated and their product taken to find the overall likelihood (as in Figure 2.5e).

The Markov models of nucleotide substitution used are time-reversible. This means that the position of a root does not affect the likelihood (which is the reason why unrooted trees only are inferred). For computational purposes, it is convenient to root the tree at an arbitrary internal node (see Figure 2.5c). To find the likelihood, the probabilities of all the possible ways in which the nucleotides at the tips could have arisen are summed (i.e., the 16 possible combinations of the two ancestral nucleotides at the two internal nodes for the example shown). Finally the likelihood, or more usually the log likelihood are calculated as in Figure 2.5e and Figure 2.5f respectively.

In practice, calculating the likelihood in this manner poses too great a computational burden. A reduction in the amount of computation required may be obtained by means of an algorithm termed *pruning* (Felsenstein, 1981), explained using the following example. Consider the tree in Figure 2.6, with branch lengths given by the v_i and the bases at each node i specified by s_i. The stationary frequency of each nucleotide j is given by π_j, while $P_{ij}(t)$ represents the transition probability that a site initially with nucleotide i has nucleotide j after a time t. As above, the likelihood of the tree is the sum of the probabilities of each way that the particular combination of bases at the external nodes could have arisen. This is given by

$$\begin{aligned} L = \sum_{s_0}\sum_{s_6}\sum_{s_7}\sum_{s_8} & \pi_{s_0} P_{s_0 s_6}(v_6) P_{s_6 s_1}(v_1) P_{s_6 s_2}(v_2) P_{s_0 s_8}(v_8) \\ & \times P_{s_8 s_3}(v_3) P_{s_8 s_7}(v_7) P_{s_7 s_4}(v_4) P_{s_7 s_5}(v_5) \, . \end{aligned} \tag{2.14}$$

This expression has 256 terms (in general, with n species there will be 2^{2n-2} terms), but by manipulating the expression slightly, it is possible to find a more efficient formulation.

If the summation signs are moved to the right, then (2.14) becomes

$$\begin{aligned} L = \sum_{s_0} \pi_{s_0} & \left\{ \sum_{s_6} P_{s_0 s_6}(v_6) \left[P_{s_6 s_1}(v_1) \right] \left[P_{s_6 s_2}(v_2) \right] \right\} \\ & \times \left\{ \sum_{s_8} P_{s_0 s_8}(v_8) \left[P_{s_8 s_3}(v_3) \right] \left[\sum_{s_7} P_{s_8 s_7}(v_7) \left[P_{s_7 s_4}(v_4) \right] \left[P_{s_7 s_5}(v_5) \right] \right] \right\} . \end{aligned} \tag{2.15}$$

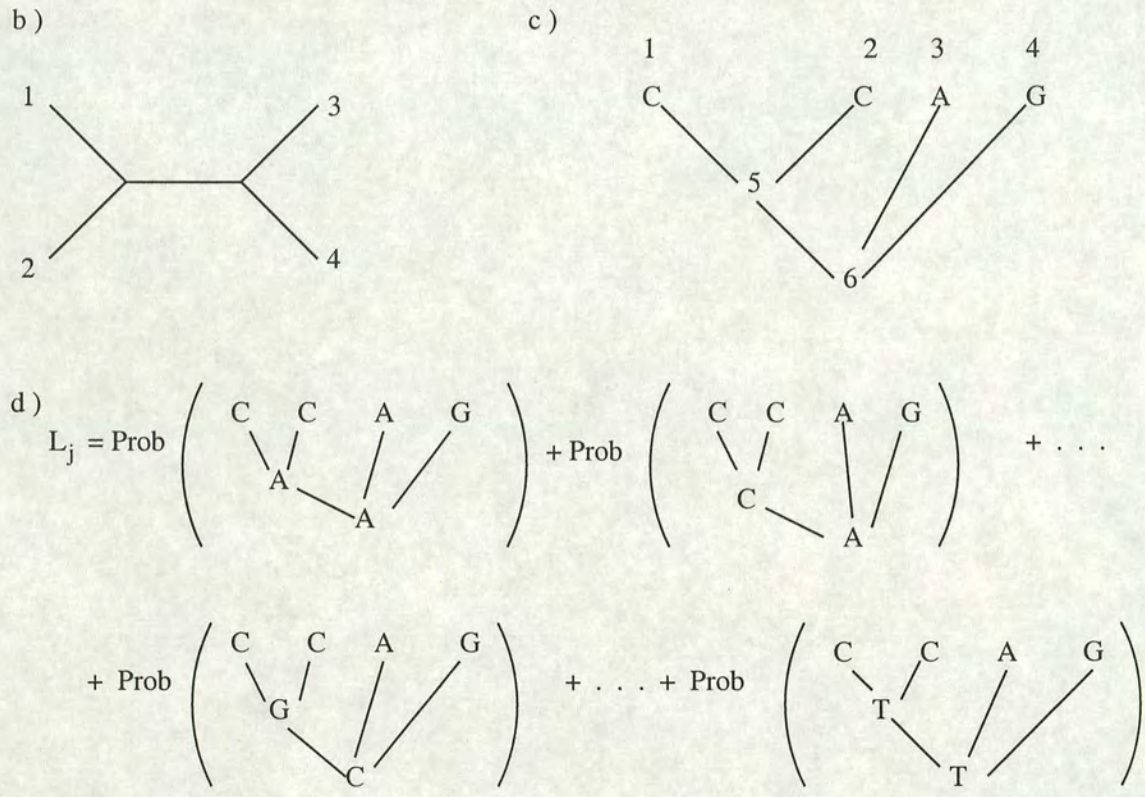
a)

1 C G G A C A C G T T T A . . . C

2 C A G A C A C C T C T A . . . C

3 C G G A T A A G T T A A . . . C

4 C G G A T A G C C T A G . . . C



$$L = L_1 \times L_2 \times \ldots \times L_N = \prod_{j=1}^{N} L_j \qquad \text{(e)}$$

$$\ln L = \ln L_1 + \ln L_2 + \ldots + \ln L_N = \sum_{j=1}^{N} \ln L_j \qquad \text{(f)}$$

Figure 2.5: Likelihood calculation for a tree. **a**: a sequence alignment. **b**: an unrooted tree for the four sequences in (a). **c**: the tree rooted at an arbitrary internal node for the nucleotides at site 7. **d**: the likelihood at site 7 is the sum of the probabilities of every possible reconstruction of the ancestral states, given some model of evolution. **e**: calculation of the likelihood for the entire sequence. **f**: calculation of the log likelihood. Redrawn from Swofford et al. (1996).

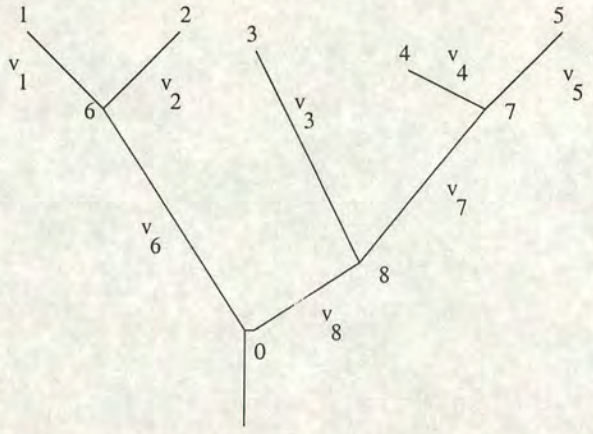


Figure 2.6: the tree used to illustrate the use of the pruning algorithm for efficient calculation of the likelihood. Redrawn from Felsenstein (1981).

The important point about the form of (2.15) is that the pattern of parentheses bears an exact relationship to the topology of the tree. Therefore, the expression can be evaluated by working outwards from the innermost parentheses. In other words, computation starts at the tips of the tree and moves downwards.

The problem may be restated in terms of conditional likelihoods. Let $L_s^{(k)}$ be the likelihood based on the data at and above node k on the tree, given that node k has nucleotide s. If k is an external node (i.e., a tip) then $L_{s_k}^{(k)} = 1$ for the nucleotide actually observed at k and zero for the other possible nucleotides. Also note that, for a node k with immediate descendants i and j,

$$L_{s_k}^{(k)} = \left[\sum_{s_i} P_{s_k s_i}(v_i) L_{s_i}^{(i)}\right]\left[\sum_{s_j} P_{s_k s_j}(v_j) L_{s_j}^{(j)}\right]. \tag{2.16}$$

Therefore, at each node k, it is straightforward to calculate $L_{s_k}^{(k)}$ for all four possible values of s_k. This process is carried out until the base of the tree is reached and $L_{s_0}^{(0)}$ has been found for each of the four possible values of s_0. Then the overall likelihood is given by

$$L = \sum_{s_0} \pi_{s_0} L_{s_0}^{(0)} \, . \tag{2.17}$$

Felsenstein (1981) termed this algorithm pruning, since each step results in the removal of two tips from the tree. It is an efficient way to calculate the likelihood of any particular tree. It does not, however, address the problem of finding the best solution.

In principle, to find the maximum likelihood solution, the branch lengths leading to the highest likelihood for all possible (unrooted) topologies are found (originally a version of the EM algorithm was used [Felsenstein, 1981] but recently the Newton-Raphson method has been used [Felsenstein and Churchill, 1996] as this significantly speeds up the computations). The tree with the highest likelihood is the required

solution. However, due to the rapid explosion in the number of possible topologies with increasing sequence number, an exhaustive search through all topologies is only possible for small data sets. Hence, it is necessary to use searching algorithms to explore the tree space for possible solutions. Heuristic methods and hill-climbing algorithms are commonly used (see Swofford et al., 1996 and references therein for more details).

2.7 Distance methods for phylogenetic tree estimation

The parsimony methods described above find solutions that minimise the amount of evolutionary change that is required to explain the data, whereas likelihood methods seek to estimate the actual amount of change that has occurred, according to a particular model of nucleotide substitution. If the rate of nucleotide substitution is high, there is an ever-increasing chance of multiple or super-imposed changes at a particular site. So, unless the actual rate of nucleotide substitution is very small, parsimony methods will underestimate the true amount of change.

Distance methods are an alternative class of methods to maximum likelihood; these also have the advantage over parsimony of using adjusted distances which correct for unseen nucleotide substitution events according to a model of nucleotide substitution such as those described in **2.5.2**. These methods are approximations to a full maximum likelihood approach since there is a loss of information by reducing two DNA sequences to a pairwise distance. Recent simulation studies have found that maximum likelihood outperforms distance methods in choosing the right tree (Kuhner and Felsenstein, 1994; Huelsenbeck, 1995). However, distance methods are considerably faster than maximum likelihood; thus they are particularly useful for large data sets.

There are two main steps when constructing a tree using a distance method: firstly, an appropriate model for the nucleotide substitution process must be chosen, and the pairwise distances between all the possible pairs of sequences in the data set must be calculated; secondly, the resulting pairwise distance matrix is used as the input into a clustering algorithm or least squares method, and a tree is then estimated. In this section, the estimation of pairwise distances is first considered, followed by a brief description of some of the algorithms in use.

2.7.1 Distance estimators based on models of nucleotide substitution

Most of the algorithms which construct phylogenetic trees from pairwise distances require additive distance measures (i.e., linear with time) for the method to work correctly. Thus, simply counting the number of distances observed between two sequences is an inappropriate measure, since distances obtained in this manner are not additive (due to unseen substitutions, the number of which increases as two sequences diverge). It must be remarked, however, that Rzhetsky and Sitnikova (1996) have discussed cases

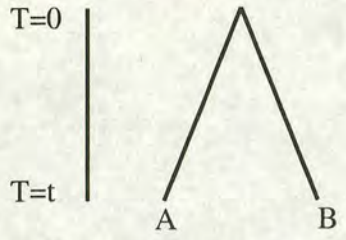


Figure 2.7: Two sequences, A and B, have evolved from a common ancestor t time units ago. Thus, the amount of change separating them is the product of $2t$ and the overall rate of change.

in which using this distance measure rather than an additive distance leads to better recovery of the tree topology (i.e., the branching pattern only).

An obvious choice of a distance measure is the average amount of change per site in the sequence. This quantity may be found by taking the product of the overall rate of evolution and the time separating the two sequences. For certain models, this may be expressed as a simple analytical formula in terms of the transition probabilities, which in turn may be estimated from the sequence data. For a substitution model to yield a simple analytical formula for the distance, Yang (1994) detailed the following two mathematical requirements which must be satisfied:

1. the eigenvectors of $\mathbf{R}$, the rate matrix, must be functions of only the nucleotide frequency parameters and thus, be free from the rate parameters;

2. the number of unknowns, not including the frequency parameters, must be the same as the number of non-zero distinct eigenvalues of $\mathbf{R}$.

Since the frequency parameters are estimated from the data, these two conditions mean that there will be as many simple equations as there are unknowns, and hence there will be a simple solution. Models which satisfy these conditions include the Jukes-Cantor, the Kimura two Parameter and the Felsenstein 81 and 84 models. Tamura and Nei's (1993) model is the most complicated model for which a simple expression is available. These conditions explain why the Felsenstein 84 model is mathematically more tractable than the HKY85 model – the rate matrix, $\mathbf{R}_{HKY85}$, for the latter model has three distinct non-zero eigenvalues, but there are only two unknowns (the transition and transversion rates).

To illustrate the procedure of finding a distance estimate, the Jukes-Cantor model is considered. For other models, the computation is similar, although necessarily more complicated due to the increased complexity of the models. An outline of the derivation of the Felsenstein 84 genetic distance estimator is given later in **6.2**.

Using (2.13), the transition probabilities for the Jukes-Cantor model are given by

$$P_{ij}(t) = \begin{cases} \frac{1}{4} + \frac{3}{4}e^{-4\alpha t} & \text{if } i = j, \\ \frac{1}{4} - \frac{1}{4}e^{-4\alpha t} & \text{if } i \neq j. \end{cases} \tag{2.18}$$

In the case of two sequences, both with nucleotide i initially, the probability that the first has nucleotide j and the second has nucleotide k after a time t is given by $P_{ij}(t)P_{ik}(t)$, assuming (as seems reasonable) that the two sequences evolve independently of each other. Consequently the probability that both sequences have nucleotide j after time t is $P_{ij}(t)^2$.

The overall rate of change for the Jukes-Cantor model from (2.5) is 3α, since each of the three possible changes happens at a rate α. If two sequences diverged from a common ancestor t time units ago, as in Figure 2.7, then the time separating the two sequences is $2t$. Hence, the distance, or the average number of nucleotide substitutions per position, is given by $2t \times 3\alpha$. Since the transition probabilities may be estimated from the data, the problem becomes that of expressing $6\alpha t$ in terms of these probabilities.

Since all changes are equally likely, the sixteen transition probabilities may be summed in two groups: those that mean a difference is observed and those that give the probability that the same nucleotide is present in both sequences. Suppose that a particular site in the ancestral sequence had the nucleotide type j. Then the probability that no change is observed between the sequences after time t, I say, is the sum of the probabilities that the nucleotide is initially j in both sequences, and after time t is k in both, for all possible values of k. Thus, I is given by

$$I = P_{jA}(t)^2 + P_{jC}(t)^2 + P_{jG}(t)^2 + P_{jT}(t)^2. \tag{2.19}$$

Using (2.18), and noting that one of the quantities in the above sum will be the square of the probability that no change is observed, while the other three will be the square of the probabilities that the chain is in a different state after time t, I may also be expressed as

$$I = \frac{1}{4} + \frac{3}{4}e^{-8\alpha t}. \tag{2.20}$$

The probability, p say, that a difference is observed may be found in a similar manner, but is more easily found by noting that $p + I = 1$. Hence,

$$p = \frac{3}{4}\left(1 - e^{-8\alpha t}\right). \tag{2.21}$$

Rearranging and taking logs of both sides yields

$$8\alpha t = -\ln\left(1 - \frac{4}{3}p\right). \tag{2.22}$$

Therefore, the distance, d_{JC}, which is equal to $6\alpha t$, is also given by

$$d_{JC} = -\frac{3}{4}\ln\left(1 - \frac{4}{3}p\right). \tag{2.23}$$

As p is the probability that a change is observed between two sequences, it may be estimated for real data by the proportion of change observed between the two sequences. This estimate is $\widehat{p} = k/n$ where k is the total number of changes observed and n is the total sequence length.

The distance estimator for the Felsenstein 81 model is obtained in a similar manner, and is given by

$$d_{F81} = -E\ln\left(1 - \frac{p}{E}\right) \tag{2.24}$$

where $E = 1 - \pi_A^2 - \pi_C^2 - \pi_G^2 - \pi_T^2$. Again $\widehat{p}$ is used in place of p to estimate a pairwise distance.

For the two parameter models, the transition probabilities must be summed in three groups: the probabilities of no change occurring; the probabilities of a particular transition occurring (sum given by P); the probabilities of a transversional event occurring (sum given by Q). For the Kimura two Parameter model the distance is given by $(\alpha + 2\beta) \times 2t$; the expression is more complicated for the Felsenstein 84 model, since it will explicitly involve the nucleotide frequences. Following a similar procedure as above, the distance under the Kimura two Parameter model is found to be

$$d_{K2P} = -\frac{1}{2}\ln(1 - 2P - Q) - \frac{1}{4}\ln(1 - 2Q), \tag{2.25}$$

while for the Felsenstein 84 model

$$d_{F84} = -2A\ln\left(1 - \frac{P}{2A} - \frac{(A-B)Q}{2AC}\right) + 2(A - B - C)\ln\left(1 - \frac{Q}{2C}\right) \tag{2.26}$$

where A, B and C are as defined in (2.12). To estimate a distance from a data set, P and Q may be replaced by their sample estimates, $\widehat{P}$ (the observed proportion of transitions) and $\widehat{Q}$ (the observed proportion of transversions).

Li and Gu (1996) discuss ways of estimating the distance using the general time reversible model specified in (2.3). They note that the difficulty in obtaining a simple analytical formula for the distance is that it depends on the eigenvalues of the rate matrix; apart from the special cases mentioned above, the eigenvalues cannot be expressed in analytical forms and thus, neither can the distance.

Let λ_k, $k = 1, \ldots, 4$ be the k^{th} eigenvalue of the rate matrix, $\mathbf{R}$, one of which will be zero, say λ_4. Define the eigenmatrix $\mathbf{U}$, with k^{th} column being the eigenvector corresponding to the k^{th} eigenvalue, and let u_{ik} be the ik^{th} element of $\mathbf{U}$. Correspondingly, let v_{ik} be the ik^{th} element of $\mathbf{V} = \mathbf{U}^{-1}$.

The number of substitutions per site (rate $\times$ time) is given by

$$\begin{aligned} d_{GTR} &= 2t \sum_{i=1}^{4} \pi_i \sum_{j \neq i} r_{ij} \\ &= -2t \sum_{i=1}^{4} \pi_i r_{ii} \end{aligned} \tag{2.27}$$

since

$$r_{ii} = -\sum_{j \neq i} r_{ij}.$$

By spectral decomposition of the rate matrix, the diagonal elements, r_{ii} may be expressed as

$$r_{ii} = \sum_{k=1}^{4} u_{ik} v_{ki} \lambda_k. \tag{2.28}$$

It has been assumed that $\lambda_4 = 0$ and substituting (2.28) into (2.27), the distance may now be expressed as

$$d_{GTR} = -2t \sum_{k=1}^{3} b_k \lambda_k \tag{2.29}$$

where

$$b_k = \sum_{i=1}^{4} \pi_i u_{ik} v_{ki} \, . \tag{2.30}$$

The distance is expressed in terms of the eigenvalues and eigenvectors of the rate matrix which cannot be estimated from the data. However, the transition probability matrix can, and since the two are related by $\mathbf{P}_{2t} = e^{2\mathbf{R}t}$, it is known that firstly they have the same eigenmatrix, $\mathbf{U}$, and secondly the eigenvalues, z_k, of $\mathbf{P}_{2t}$ are related to those of $\mathbf{R}$ by $z_k = \exp\{2t\lambda_k\}$. Because $\lambda_4 = 0$, $z_4 = 1$. Expressing λ_k in terms of z_k, the distance (2.29) may be rewritten as

$$d_{GTR} = -\sum_{k=1}^{3} b_k \ln z_k \, . \tag{2.31}$$

To find a distance estimate for two species x and y, the transition probability matrix must be estimated. To do this, the 4×4 matrix $\mathbf{F}^s_{xy}$ is formed, with the i^{th} diagonal entry given by N_{ii}/N and the ij^{th} off-diagonal entry given by $(N_{ij} + N_{ji})/2N$, where N_{ij} is the number of sites having nucleotide i in species x, and nucleotide j in species y, and N is the total number of sites. This matrix is an estimate of the transition probabilities, and its eigenvalues and eigenvectors may be used in (2.31).

Lanave et al. (1984) and Rodriguez et al. (1990) also consider ways of formulating a distance measure for the general time-reversible model. Their algorithms give essentially the same results.

2.7.2 Estimates of the variance and confidence intervals for distance estimators

Sometimes an estimate of the error of these estimators is required. A popular and simple method to estimate the variance is the delta method. If $m(\mathbf{V})$ is a function of a statistic $\mathbf{V}$, with known variance-covariance matrix Σ, then the variance of $m(\mathbf{V})$ may be approximated by

$$\mathrm{Var}[m(\mathbf{V})] \approx \frac{\partial}{\partial \mathbf{v}^T} m(\mathbf{v})\, \Sigma\, \frac{\partial}{\partial \mathbf{v}} m(\mathbf{v})\Big|_{\mathbf{v}=\boldsymbol{\mu}} \tag{2.32}$$

where $\boldsymbol{\mu}$ is the expectation vector of $\mathbf{V}$. This was first introduced into the phylogenetic literature by Kimura and Ohta (1972); they use this method to find an approximation to the variance of the Jukes-Cantor distance estimator. They note that the distance estimator depends on p, the probability of observing a difference and this is estimated by the sample statistic $\widehat{p} = k/n$ where k is the number of differences observed and n is the sequence length. Clearly k is an observation from a binomial distribution and thus, the variance of $\widehat{p}$ may be found.

For more complicated models, the sample statistic is comprised of observations from a multinomial distribution; the algebra is more tedious, but the procedure is essentially the same. This is discussed in some more detail in Chapter 6, where improvements in the calculation of confidence intervals are developed.

If confidence intervals are required for the distance estimator, an assumption of normality may be made to allow the calculation of these intervals. Such an assumption is questionable, especially in the case of short sequences and/or large distances since it is well known that the sampling distribution of these distance estimators is biased and skewed to the right. Since the sample statistic is a sum of independent random variables (the observation at each site), by the Central Limit Theorem, the sampling distribution should approach normality as the sequence length increases. However, it is possible that this may not occur for many sequence lengths and distances used in practice.

Other methods may be used to approximate the variance, for example non-parametric bootstrapping. This was introduced by Felsenstein (1985) as a means of testing the statistical significance of clusters in a phylogenetic tree, but it may also be used to approximate the variance of a pairwise distance measure. It involves generating many new samples of the same length as the original alignment by resampling the sites in the alignment with replacement. The variability in the resulting bootstrap samples should reflect the variance of the distance estimator.

Another possibility is interval estimation, which was recently suggested by Andrieu et al. (1997). They use this procedure to find the exact confidence intervals for the Jukes-Cantor and Kimura two Parameter models. To illustrate this method, consider

the estimation of a confidence interval for a Jukes-Cantor distance, $\widehat{d}$.

Suppose that k is an observation from a Binomial distribution with parameters n, the number of trials and p, the probability of success. Let $d = f(p)$ be a function of p. Then $p \in [\underline{p}, \overline{p}]$ is equivalent to $d \in [f(\underline{p}), f(\overline{p})]$. Therefore, the problem is to find the values of $\underline{p}$, $\overline{p}$ to yield the desired confidence interval.

In practice, the sampling distribution of $\widehat{p}$ is usually well approximated by a normal distribution and hence, normal sampling theory is used to find confidence intervals. This is seen later in Chapter 6. However, when n or p are small the approximation may not be sufficiently accurate. Problems arise if the observed number of successes is zero; sampling theory is unhelpful as the variance is estimated as zero. In this case, the exact confidence interval for $\widehat{p}$ would be more useful.

There are two steps in the calculation of exact intervals for the estimator of the probability from a binomial distribution. Firstly, let K be a binomial random variable with parameters n and p. For any fixed value of p, the distribution of K is known, and the functions $\underline{K} = \underline{K}(p, n, \alpha)$ and $\overline{K} = \overline{K}(p, n, \alpha)$ may be defined as the largest integer $\underline{K}$ and the smallest integer $\overline{K}$ such that

$$\text{Prob}(K \geq \underline{K}) = \sum_{i=\underline{K}}^{i=n} \binom{n}{i} p^i (1-p)^{n-i} \geq 1 - \frac{\alpha}{2}$$

and

$$\text{Prob}(K \leq \overline{K}) = \sum_{i=0}^{i=\overline{K}} \binom{n}{i} P^i (1-p)^{n-i} \geq 1 - \frac{\alpha}{2}. \tag{2.33}$$

These functions define the smallest interval in which K lies with probability greater than or equal to $1 - \alpha$.

For the second step, suppose now that k successes are observed. The aim is now to find the set of all possible values of p, $\widetilde{p}$, such that k will lie in the corresponding intervals $[\underline{K}(\widetilde{p}, n, \alpha), \overline{K}(\widetilde{p}, n, \alpha)]$. Let $\underline{p} = \underline{p}(k, n, \alpha)$ and $\overline{p} = \overline{p}(k, n, \alpha)$ be the lower and upper bounds respectively of this range of values. Then these bounds may be found by solving

$$\sum_{i=k}^{i=n} \binom{n}{i} \underline{p}^i (1-\underline{p})^{n-i} = \frac{\alpha}{2}$$

and

$$\sum_{i=0}^{i=k} \binom{n}{i} \overline{p}^i (1-\overline{p})^{n-i} = \frac{\alpha}{2}. \tag{2.34}$$

In the case of DNA sequences, n is the sequence length, p is the probability of observing a change of nucleotide at a particular site in the sequence and k is the

observed number of changes separating the two sequences. The above method may be used to find exact confidence intervals for $\widehat{p}$, the estimator of p. The Jukes-Cantor distance function, $d = -3/4 \ln(1 - 4p/3)$, may then be used to transform the upper and lower bounds for $\widehat{p}$ to yield those for $\widehat{d}$.

Similar steps may be implemented to find the exact confidence intervals for the Kimura two Parameter model. The computations are, of course, more complicated as the distance estimator is a function of observations from a multinomial distribution. There is also the added problem that it is necessary to assume that the exact value of the transition-transversion ratio is known; in practice this is very unlikely to ever be the case. Note that similar computations may be carried out for the one and two parameter models which allow for non-equal base composition (the Felsenstein 81 and 84 models respectively).

As mentioned above, this procedure is useful in the case of small amounts of change since methods based on sampling theory will not be very helpful. In general however, interval estimation is a tedious way of estimating confidence intervals. Equations (2.34) and the corresponding ones for the Kimura two Parameter model (see Andrieu et al., 1997) cannot be solved analytically, involving instead a certain amount of iterating to find the solution. In addition, for the two parameter models, the value of the transition-transversion ratio must be assumed to be known. Therefore, there is still scope to improve inferences on genetic distances.

2.7.3 Other distance estimators

Many of the distance estimators above are biased, and sometimes cannot be applied to the data since they involve logs (when the argument of the log is negative, the formulae cannot be used). Tajima (1993) and Rzhetsky and Nei (1994) have looked at ways of dealing with these problems using Taylor series expansions of the log term. The resulting formulae may always be applied, and often give almost unbiased estimators.

As an illustration, consider the formula for the distance estimator for the Jukes-Cantor model, given in (2.23). Using a Taylor series expansion, (2.23) may also be expressed as

$$\begin{aligned} d_{JC} &= p + \frac{p^2}{2E} + \frac{p^3}{3E^2} + \dots \\ &= \sum_{i=1}^{\infty} \frac{p^i}{iE^{i-1}} \end{aligned} \tag{2.35}$$

An unbiased estimator of p^i is $k^{(i)}/n^{(i)}$, $i \leq k$, where

$$\begin{aligned} x^{(i)} &= x(x-1)(x-2)\dots(x-i+1) \\ &= \frac{x!}{(x-i)!} \end{aligned}$$

[since k is a binomial random variable with parameters n (sequence length) and p (probability of observing a difference)]. Ignoring terms higher than the k^{th} order in (2.35) suggests the distance estimator

$$\widehat{d} = \sum_{i=1}^{k} \frac{k^{(i)}}{iE^{i-1}n^{(i)}} \tag{2.36}$$

with expectation

$$E(\widehat{d}) = \sum_{i=1}^{k} \frac{p^i}{iE^{i-1}}.$$

Thus, (2.36) should give an almost unbiased estimate when p is not close to E. Consequently, it will be useful for eliminating the bias for short distances, but will not be a good estimator when the evolutionary distances are large, even though it will always be possible to use (2.36) unlike (2.23).

Goldstein and Pollock (1994) considered alternative ways of estimating a linear distance, making the same assumptions as the Kimura two Parameter model (equal base composition; different rates for transitions [α] and transversions [β]). They estimated the number of transitional ($2\alpha t = S_t$) and transversional ($4\beta t = V_t$) changes from the data. Since $\overline{V}_t = V_t\alpha/(2\beta)$ is equal to $2\alpha t$, this may also be used to estimate the amount of transitional change. The best evolutionary distance (linear expectation, minimal variance) based on these measures of transitional and transversional change may then be obtained using generalised least squares. This is the value of D which minimises

$$\sum_{i=1}^{2}\sum_{j=1}^{2} w_{ij}(D - x_i)(D - x_j), \tag{2.37}$$

where $x_1 = S_t$ and $x_2 = \overline{V}_t$. The weights, w_{ij}, are the inverse of the variance-covariance matrix of the distance estimators. Goldstein and Pollock (1994) give expressions for the elements of this matrix, as well as the resulting distance estimator, which they term LSD. Note that LSD estimates $2\alpha t$, the amount of transitional change. Simulations comparing LSD to the Kimura two Parameter and Jukes-Cantor distance estimators suggest that LSD is indeed an improved distance estimator.

In **2.2**, the existence of rate variation in the nucleotide substitution process was discussed. It is very important to take this into account, particularly in protein-coding DNA sequences where the third codon position may be evolving at a considerably faster rate than the first two positions. One way of doing this is to use gamma mixing, and was first used in the phylogenetic literature by Jin and Nei (1990) for the Kimura two Parameter model. They assume that the transition-transversion ratio (τ) is fixed, and that the overall rate of nucleotide substitution ($\lambda = \alpha + 2\beta$ from equation 2.6) varies

according to a gamma distribution with shape a where $a = \overline{\lambda}^2/\mathrm{Var}(\lambda)$. Using the well-known result that if $Y \sim \Gamma(m,n)$, then $cY \sim \Gamma(m,n/c)$, it can be shown that $\beta \sim \Gamma(a,d)$ where $d = cb$, c a constant ($c = 2\tau + 2$) and $\alpha + \beta \sim \Gamma(a,f)$ where again $f = kb$ ($k = 2\tau + 1$).

Since the rates of substitution vary from site to site the proportion of changes per site, averaging over the rates, must be found. Without rate variation, the number of transversions is given by $Q = 1/2 - 1/2e^{-8\beta t}$. Averaging over rate yields

$$\begin{aligned}\overline{Q} &= \int_0^\infty Q f(\beta) d\beta \\ &= \frac{1}{2} - \frac{1}{2}\int_0^\infty e^{-8\beta t}\frac{d^a}{\Gamma(a)}\beta^{a-1}e^{-d\beta}d\beta \\ &= \frac{1}{2} - \frac{1}{2}\left[\frac{d}{d+8t}\right]^a.\end{aligned}$$

Multiplying above and below by a/d and noting that $\overline{\beta} = a/d$,

$$\overline{Q} = \frac{1}{2} - \frac{1}{2}\left[\frac{a}{a+8\overline{\beta}t}\right]^a. \tag{2.38}$$

$\overline{P}$ may be calculated in a similar fashion:

$$\overline{P} = \frac{1}{4} - \frac{1}{2}\left[\frac{a}{a+4(\overline{\alpha}+\overline{\beta})t}\right]^a + \frac{1}{4}\left[\frac{a}{a+8\overline{\beta}t}\right]^a. \tag{2.39}$$

Using (2.38) and (2.39) it can be shown that

$$2\overline{\beta}t = \frac{a}{4}\left[\left(1-2\overline{Q}\right)^{-1/a} - 1\right]$$

and

$$2(\overline{\alpha}+\overline{\beta})t = \frac{a}{2}\left[\left(1-2\overline{P}-\overline{Q}\right)^{-1/a} - 1\right].$$

Therefore, the distance is given by

$$\begin{aligned}d &= 2\overline{\alpha}t + 4\overline{\beta}t \\ &= \frac{a}{2}\left[\left(1-2\overline{P}-\overline{Q}\right)^{-1/a} + \frac{1}{2}\left(1-2\overline{Q}\right)^{-1/a} - \frac{3}{2}\right].\end{aligned} \tag{2.40}$$

To estimate this distance from a data set, the quantities $\overline{P}$ and $\overline{Q}$ are replaced by their estimates, $\widehat{P}$ and $\widehat{Q}$ (the observed proportion of transitions and transversions) respectively.

Other possible distance measures are the LogDet (Steel, 1994; Lockart et al., 1994) or the paralinear distances (Lake, 1994). This is a transformation yielding additive distances (see **2.7.4**) under a wider set of models; any Markov model of nucleotide substitution is feasible, as long as the sites evolve independently and identically, and the rates of substitution are equal across sites. To find the LogDet distance estimate

for a pair of sequences, the matrix $\mathbf{F}_{xy}$ is found. The ij^{th} entry of this matrix is given by N_{ij}/N (N_{ij} is the number of sites where the first sequence has nucleotide i and the second has nucleotide j while N is the total number of sites). The distance is then estimated as

$$-\ln\left[\det \mathbf{F}_{xy}\right]. \tag{2.41}$$

The distance estimates described previously depend on commutative multiplication of matrices which greatly restricts the type of model which may be used. However, since (2.41) uses determinants, multiplication of these is always commutative, so more general models are allowed. The only conditions are that the determinant of $\mathbf{F}_{xy}$ is not 0, 1 or -1. (2.41) can accommodate changing base composition when finding the pairwise distances between a set of species; this is something which can seriously mislead phylogenetic tree estimation methods when standard distance estimates are used.

One drawback to the LogDet distance is that it does not estimate the number of substitutions which have occurred. It is possible to modify the distance estimate for some special cases to yield an estimate of the number of substitutions which have occurred. Essentially, these special cases comprise of the types of models described in **2.5.2**. Indeed, the procedure described by Li and Gu (1996) may be restated in matrix terms and depends on the trace of the log of $\mathbf{F}^s_{xy}$, which is equivalent to finding the log of the determinant of $\mathbf{F}^s_{xy}$.

2.7.4 Properties of pairwise distance estimates

Before a review of some phylogenetic tree estimation methods is given, some properties of distance measures are defined. Most methods require the distances to be *additive*, i.e., the sum of the branch lengths joining any two taxa is equal to the distance between them. Such distances must satisfy the *four-point metric condition* (Buneman, 1971): for any four taxa A, B, C and D,

$$d_{AB} + d_{CD} \leq \max\{d_{AC} + d_{BD}, d_{AD} + d_{BC}\}. \tag{2.42}$$

This has a simple meaning: of the three sums of distances, $d_{ij} + d_{kl}$, where i, j, k and l are all distinct, one of these must be as small, or smaller than the other two, and these other two must be equal. For real data, the pairwise distances are very unlikely to be additive, even if the model of nucleotide substitution was exact (which would only be the case for simulated data). This is due to the fact that there is only a finite amount of data, so stochastic errors will cause the distances to deviate from additivity.

An even more restrictive property of distances is the *ultrametric* property. This requires the three point condition to be met: for any three taxa A, B and C,

$$d_{AC} \leq \max\{d_{AB}, d_{BC}\}. \tag{2.43}$$

This is equivalent to saying that two of the pairwise distances between two taxa are equal, and at least as large as the third. Ultrametric distances will fit an additive phylogenetic tree, with the additional feature that it can be rooted so that all of the taxa are equidistant from the root (i.e., the tips of the tree all finish at the same vertical line in a dendrogram). This is equivalent to saying that a *molecular clock* must exist (the sequences in the data set all evolve at the same rate). Due to stochastic error, it is very unlikely that estimated distances will be ultrametric, even if the molecular clock hypothesis is true for a particular data set.

Some of the available distance methods are discussed now. These may be split into two groups: the algorithmic type which produce one answer only; and those with an optimality criterion, which means that a search of tree space must be carried out to find possible solutions.

2.7.5 Algorithmic phylogenetic tree estimation techniques using pairwise distance data

UPGMA

UPGMA, or Unweighted Pair Group Method with Arithmetic Averages was one of the first distance methods to be suggested (Sneath and Sokal, 1973), and for a time was widely used. This is essentially average linkage cluster analysis and requires ultrametric distances. Distances which do not satisfy this criterion will generally lead to incorrect estimates of a phylogenetic tree using UPGMA. Simulation studies have suggested that UPGMA is inefficient and confirmed that it is extremely sensitive to departures from ultrametric distances (Huelsenbeck, 1995), often leading to very wrong estimates of the underlying phylogenetic tree. This has been partly responsible for the early unpopularity of distance methods.

Neighbor Joining

The Neighbor Joining method (Saitou and Nei, 1987) could be described as a type of cluster analysis, which allows for unequal rates of evolution along the branches of the phylogenetic tree. It does this by constructing a transformed distance matrix at each step in the analysis; the transformation adjusts the distance between each pair of nodes on the basis of the mean divergence from all other nodes. For details, see Avise (1994). Once this matrix has been obtained, the two nodes separated by the smallest distance are joined. Simulation studies (Kuhner and Felsenstein, 1994) suggest that this method performs reasonably well in practice, although there is the problem, particularly for larger data sets, that once two nodes have been joined, they cannot be unjoined.

Saitou and Nei (1987) have shown that the step which estimates the branch lengths between two neighbours is the unweighted least squares estimate (Cavalli-Sforza and Edwards, 1967, see below) for a tree with nodes i and j as neighbours and with all

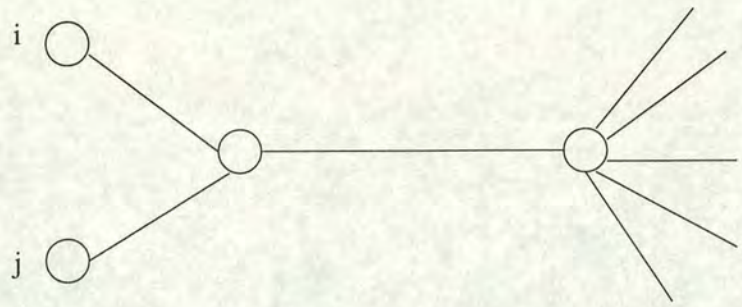


Figure 2.8: The branch length estimation of the Neighbor-Joining algorithm between two neighbours is equivalent to the unweighted least squares estimation of the branch lengths leading to nodes i and j for this type of tree.

other tips branching out from a multifurcating node (see Figure 2.8). Therefore, it may be viewed as an approximation to the least squares solution (Felsenstein, 1997). In the simulation study described by Kuhner and Felsenstein (1994), Neighbor-Joining was found to perform almost as well as least squares; its success suggests that the estimate of branch lengths between two neighbours is not highly sensitive to the resolution of the relationships between the taxa involved in the multifurcation.

2.7.6 Estimating phylogenetic trees using least squares

A class of estimation methods involves minimising the differences between the estimated tree distances and the observed distances from the pairwise distance matrix. This is done by minimising an objective function of the form

$$E = \sum_{i=1}^{T-1} \sum_{j=i+1}^{T} w_{ij} |d_{ij} - p_{ij}|^{\alpha} \tag{2.44}$$

where E is the error in fitting the distance estimates to the tree

T is the number of species

w_{ij} is the weight applied to the branch lengths between sequences i and j

d_{ij} is the estimate of the pairwise distance between sequences i and j

p_{ij} is the predicted distance between sequences i and j, from the tree

α is either 1 or 2.

(Swofford et al., 1996). The value of α is often chosen to be 2, which places this method into the least squares group, the class of methods considered here. In addition, values for the $\{w_{ij}\}$ must be chosen. These reflect the magnitude of error in the distance estimates. If it is believed that they are all subject to the same magnitude of error, then $w_{ij} = 1$ is appropriate (Cavalli-Sforza and Edwards, 1967) while if the estimates are assumed to be uncertain by the same percentage, $w_{ij} = 1/d_{ij}^2$ is a reasonable choice (Fitch and Margoliash, 1967).

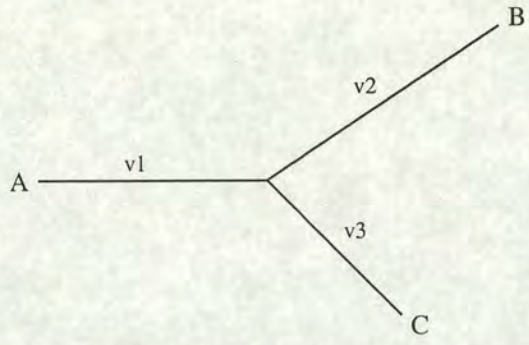


Figure 2.9: An example of a tree for three species, A, B and C, with branch lengths as shown.

If the observed distances are additive for a particular topology then exact branch lengths may be fitted to the data. Otherwise (as is usually the case), the objective is to minimise E, the discrepancy between the observed and the estimated distances. A particular tree topology is chosen. The object now is to find the branch lengths. To illustrate how this may be carried out, consider the problem of finding the branch lengths of the simple tree in Figure 2.9, with observed pairwise distances of d_{AB}, d_{AC} and d_{BC}. The branch lengths may be found by solving the system of equations:

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} p_{AB} \\ p_{AC} \\ p_{BC} \end{pmatrix}$$

$$\mathbf{Av} = \mathbf{p} \tag{2.45}$$

where $\mathbf{v}$ is the matrix of branch lengths and $\mathbf{p}$ is the vector of predicted distances between the sequences. The matrix $\mathbf{A}$ specifies the linear combinations of $\mathbf{v}$ which yield each of the elements of $\mathbf{p}$.

In the case of additive distances, linear algebra may be used to obtain the solutions. The vector $\mathbf{p}$ is replaced by $\mathbf{d}$, the vector of the observed pairwise distances. If $w_{ij} = 1$ then $\mathbf{v} = (\mathbf{A}^T\mathbf{A})^{-1}(\mathbf{A}^T\mathbf{d})$, while if $w_{ij} = d_{ij}^{-2}$ then $\mathbf{v} = (\mathbf{A}^T\mathbf{W}\mathbf{A})^{-1}(\mathbf{A}^T\mathbf{W}\mathbf{d})$, where W is a $T(T-1)/2 \times T(T-1)/2$ matrix with diagonal elements equal to the weights associated with each pairwise distance, and all off-diagonal elements equal to zero. For non-additive distances, algorithms must be used to obtain the least squares estimates (Felsenstein, 1997).

The best tree is the one which minimises E, so for non-additive distances, the idea is that the best set of branch lengths are found for each topology, and the overall best tree is the one which minimises E. For large data sets, it is impossible to find the best branch lengths for all topologies as the number of trees is too great. Hence, heuristic search algorithms must be employed to search the tree space for good solutions. More details may be obtained in Swofford et al. (1996) and references therein.

Least squares methods have been found to perform quite well in various simulation studies (Kuhner and Felsenstein, 1994). For four species trees, such as that in Figure 2.4, least squares estimates the correct tree over much of the parameter space, performing quite well for some trees with branch lengths in the Felsenstein zone (Huelsenbeck, 1995). One drawback is that negative branch lengths may result from the minimisation of E in (2.44), but it is possible to include the constraint that branch lengths must be greater than or equal to zero.

Phylogenetic trees are often used for subsequent analyses so other questions will need to be answered. These vary from model diagnostics (does the chosen model of evolution fit the data reasonably well) to the confidence in the tree (how significant are the branching patterns observed) to the comparison of one hypothesis to another (is tree A significantly different to tree B). Some tests which have been developed to address these questions are examined in the next section.

2.8 Statistical tests

As implied above, statistical tests in the area of phylogenetics may be, by and large, divided into three groups: those which test the fit of a model; those which assess the confidence in a particular tree; and those which directly compare two trees to each other.

For distance and likelihood inference, it is important to select an adequate model of nucleotide substitution. Goldman (1993a) suggests using a likelihood ratio to test one model (model 0) with n parameters against a more complex version (model 1) with m parameters where $n < m$. The proposed test statistic is $\delta = 2(\ln L_1 - \ln L_0)$ where L_i is the likelihood under model i. It might be expected that this statistic would have a χ^2 distribution with $m - n$ degrees of freedom. However, Goldman (1993a) observes that this approximation sometimes does not hold for phylogenetic problems, so he suggests simulating a large number of data sets under the null hypothesis (model 0) and finding the value of δ for each simulation. This yields a distribution for δ if the null hypothesis that model 0 provides an adequate description of the data is true. Hence, the significance of the observed value may be assessed.

This principle of simulating data under a null hypothesis to assess the significance of a test statistic is generally referred to as *parametric bootstrapping* in the phylogenetic literature, a very useful technique in modern phylogenetic analysis (for example, see Huelsenbeck et al., 1996). Huelsenbeck and Bull (1996) use parametric bootstrapping in a test for heterogeneous regions in DNA sequences (e.g., regions with different phylogenies or regions evolving under different conditions). Standard models will provide a poor fit to such data sets.

Goldman (1993b) also considers specific deviations from models of evolution. For example, studies have shown that, for some data, allowing for invariable sites (positions in an alignment which cannot change) makes a significant improvement in the fit of a model. He develops a test to see if the number of constant sites observed in a sequence is greater or less than expected, employing a normal approximation. Other properties examined include the number of different permutations of the four nucleotides observed at the positions in a multiple alignment.

Rzhetsky and Nei (1995) have developed tests which examine the performance of nucleotide substitution models for a data set using properties of the model. For example, to assess the suitability of the Jukes-Cantor model to a particular data set, the property of this model that the expected number of transversional differences between two sequences is twice that of the number of transversional differences is used to define a test statistic, with known variance. Significance of the observed value may be assessed using a normal distribution.

Resampling methods have been used to assess the confidence in certain branching patterns in a tree, in particular, the non-parametric bootstrap (Felsenstein, 1985, 1988), so called to distinguish it from the model-based parametric bootstrap. Essentially this generates a large number of pseudo data sets by sampling the columns of the data set with replacement. Thus, some columns of the multiple alignment may be sampled several times, while others will not be present at all in the pseudo data set. Trees may then be inferred from each of these data sets. If a particular group is present in a large number of these trees (e.g., around 95%) then the group may be said to be significantly supported. Bootstrapping may be used with any phylogenetic inference method, though for large data sets, its use with maximum likelihood will often be too slow to be of practical use.

To avoid the problem of multiple tests, it is important to decide on a hypothesis before carrying out an analysis. For example, it might be of interest to see if a particular group of sequences is monophyletic (i.e., of common descent and thus separated from the rest of the sequences in a tree). A consensus tree is constructed from the trees estimated from each bootstrap replicate, and the bootstrap support for each branch is shown (i.e., the number of trees which have the two groups separated by that branch as two distinct groups). It is then simple to find the support for the particular group of interest.

The bootstrap values are difficult to interpret: does 95% bootstrap support correspond to 95% confidence that a group is, indeed, monophyletic? There has been some work done on the properties of these bootstrap values. Many believe that when the phylogenetic tree inference method used is consistent, high bootstrap values tend to underestimate the confidence in a particular cluster, whereas the opposite appears to be

true for low values – they overestimate the confidence. The extent of this bias seems to depend on factors such as the number of species in the tree, the length of the sequences, and the locations of the internal branch being assessed for significance (Swofford et al., 1996, and references therein). However, Efron et al. (1996) recently suggested that this apparent bias was a result of the greater variance of the bootstrap estimates about the true tree, implying that there is no systematic bias in the estimates.

One disadvantage of non-parametric bootstrapping is that it cannot detect an incorrectly inferred tree topology. For example, an implausible method of tree inference might be to group sequences in alphabetical order of their names. All bootstrap replicates would produce the same tree which is highly unlikely to be the correct tree. Thus, non-parametric bootstrapping can lure the user into a false sense of security.

A researcher may have a particular hypothesis about the evolution of a data set, which translates into a certain branching pattern. However, when they estimate the best tree for the data set, they may find the branching pattern is different. But is it significantly different? Kishino and Hasegawa (1989) proposed a test which may be used in such a case, using likelihoods. They compare a particular topology (H_1) to the estimated one (H_0) using the posterior probability of observing H_1 if H_0 is true, this probability depending on the difference in the log likelihoods. The variance of this difference may be estimated from the likelihoods at each site. Since the log likelihoods at each site are assumed to be independent, identically distributed random variables, the log likelihood for each model, and consequently the difference in log likelihoods should be approximately normal. Thus, a confidence interval for the posterior probability of H_1 may be found and this may be used to assess if H_1 is significantly worse than H_0. Since this test does not use bootstrapping, it is quick to carry out. It does require the use of likelihoods, but for large data sets, trees may be estimated using other, faster methods and then their site likelihoods may be evaluated.

Parametric bootstrapping may be used to assess the evidence supporting a particular hypothesis about a tree topology. This is best illustrated by means of an example used by Hillis et al. (1996) for the same purposes. A dentist who was HIV positive was suspected of having infected some of his patients. Constructing phylogenetic trees based on HIV samples from the dentist, his patients and from other sources in the local area allowed the investigation of this charge. One interesting fact arose in the study: one patient had two strains of HIV which appeared to have separate origins in the estimated phylogenetic tree. Since this patient had multiple risk factors for HIV, this suggests the possibility of multiple infection, which would be of considerable interest to epidemiologists.

To assess the evidence for this hypothesis, the null hypothesis was chosen to be that the patient was not infected from multiple sources. Thus, the phylogenetic tree had

all branches as before, except that the two HIV sequences from the patient clustered together. The best tree for this topology was found. Then 100 replicate data sets were simulated according to this tree and the model of evolution which had been used. The difference in the log likelihoods between the null hypothesis tree compared to the optimal tree (if different, the result of random errors) were recorded and used to form a distribution for this difference. The actual observed difference was far greater than any from this distribution, and thus it was concluded that the patient was infected from multiple sources.

One argument against the use of parametric bootstrapping in this manner is that the results might be sensitive to the choice of model of nucleotide substitution. However, the procedure may be repeated using different models of substitution to assess the sensitivity of the results (analogous to an investigation into the dependency of the results of a Bayesian analysis on the choice of prior). Hillis et al. (1996) state that limited studies have suggested that the test is robust to changes in the model of evolution.

Chapter 3

A Review of Tests for Recombination

A large part of the work in this thesis deals with detecting evidence of a phenomenon called *recombination* in DNA data sets. Hence, a review of existing methods for inferring the presence of recombination is given here. The chapter opens by describing the recombination process and its biological importance before discussing tests for recombination.

3.1 Description of recombination

Recombination is a genetic process that results in the exchange or transfer of DNA subsequences between two DNA sequences. In species with two pairs of chromosomes (e.g., humans), recombination events involve the exchange of DNA subsequences between chromosomes and produce an offspring whose DNA is a mosaic of the DNA from the parents. Recombination in bacteria involves the transfer of DNA subsequences from one organism to another. Bacteria have one large chromosome (the circular genome shown in Figure 3.1a). If a DNA subsequence from another bacterium is present in the environment of a bacterium, it can remove the corresponding piece of DNA from its genome and replace it with the foreign genetic material as shown in Figure 3.1b. Thus, within species, recombination is a process which mixes the genetic material and increases variation.

Recombination can result in the horizontal transfer of DNA from one bacterial or viral species or strain to another (what constitutes a distinct species is often not clear cut with bacteria and viruses. Therefore, the term *strain* is often used; this can be thought of as the equivalent of species, but with less clear cut species boundaries).

Recombination is an important source of variation in many bacteria and viruses. Robertson et al. (1995) note that recombination in strains of HIV-1 is relatively frequent and appears to be a significant source of new variation observed in HIV-1. Recombination may also be an important source of genetic variation in strains of the Hepatitis

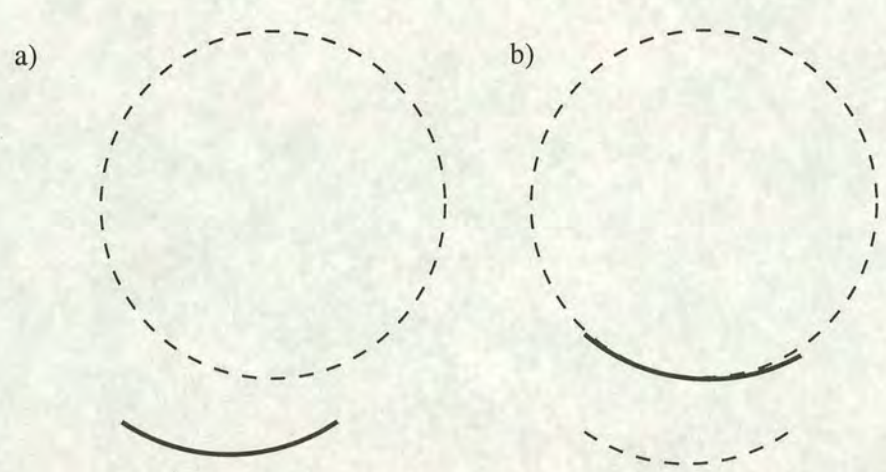


Figure 3.1: an illustration of recombination in bacteria. The circle represents the genome of a bacterium, the solid arc depicts some genetic material in the environment. **a**: before the recombination event. A piece of foreign DNA is in the bacterium's environment. **b**: the bacterium has included the foreign DNA in place of its own in its genome.

B virus. Bollyky et al. (1996) examined 25 strains of the virus and found two recombinant sequences (i.e., mosaic sequences, containing DNA from different sources), both of which came from a geographic region where multiple genotypes are known to coexist. A further example of recombination in bacteria is the argF gene of *Neisseria Meningitidis*. Zhou and Spratt (1992) found two regions of high diversity in this gene, and identified one as a recombinant.

The detection of recombination is very important for many applications. For example, in AIDS research potential vaccines will often be developed for particular strains of the virus. If it is known that a particular strain is actually a mosaic of established types, then a potential vaccine could be tailored accordingly. Recombination is also the vehicle through which many disease-causing bacteria acquire resistance to antibiotics, so again it is important to be able to detect instances of its occurrence.

3.2 Using polymorphic sites to detect recombination

Various methods have been proposed which use polymorphic sites in a DNA sequence alignment to detect recombination. A site in an alignment is said to be polymorphic if there is more than one nucleotide type among the sequences at that position. Polymorphic sites include those sites classed as informative under the parsimony criterion. Informative sites provide support for particular branching patterns. For example, suppose the nucleotides at a particular site in a four species data set were $AGAG$. Then this site is informative since it suggests that species 1 and 3 cluster together. A site with nucleotides $AGGG$ is not informative, however, as this does not suggest any partitions of the data beyond the trivial one (species 1 v species 2, 3 and 4).

Stephens (1985) was one of the earliest developers of statistical techniques to detect recombination. He takes an alignment of a small number of DNA sequences and considers those polymorphic sites which generate a split in the data set (i.e., there are exactly two nucleotide types at a site and the DNA sequences may be partitioned according to which of them has the first nucleotide type and which has the second). For example, if site j in a five-sequence data set has nucleotides $GAGAA$, then the data set may be partitioned into sequences 1 and 3, with the other three forming the other set. Partitions into three or four sets are not considered as it is usually sufficient to consider two-set partitions only.

For any particular split into two subsets, those polymorphic sites which support this split are considered. Stephens (1985) develops tests to see if s sites supporting the partition are significantly clustered. If this is the case, then it suggests that a recombination event has occurred.

One problem with these tests is that it is difficult to apply them to larger data sets. Especially in the case of high levels of polymorphism, finding an informative split may be difficult. Multiple comparisons also causes problems for larger data sets. If a data set contains n sequences, then there are $2^{n-1}-1$ possible splits. For eight sequences, these leads to 127 possible partitions. Therefore, some splits are likely to have significant non-random clustering by chance.

A further drawback is that, while the tests can detect recombination events, they do not find the location of the breakpoints. If the aim of an analysis is to infer the phylogenetic relationships within a set of species, then knowing the limits of recombination events is very important. Sawyer (1989) noted that these tests had another shortfall: they only correct for regions with high rates or low rates of substitution along a sequence by deleting segments with no polymorphic sites. With moderate levels of polymorphism, it would be desirable to have a more sensitive way of allowing for variable rates of nucleotide substitution. Therefore, he proposed a test based on fragments of DNA sequences, which does take account of variable mutation rates.

Given a set of n aligned DNA sequences, a site is said to be both silent and polymorphic if the nucleotides at this site are not identical in all the sequences but the amino acids encoded by the site's codon in the sequences are identical. Suppose there are, in total, s silent polymorphic sites in the alignment. If two of the sequences are then compared, they will differ at $d < s$ silent polymorphic sites. These sites partition the DNA sequences into $d+1$ subsets, called *fragments*. A *condensed fragment* is the set of all the silent polymorphic sites in the fragment, its length, x_i, being the number of such sites. Clearly, the sum of the lengths of all the condensed fragments is given by $\sum x_i = s - d$. The sum of the squares of the condensed fragment lengths, $SSCF$, is defined as the sum of x_i^2 over all d_k+1 fragments over all $n(n-1)/2$ pairs of sequences,

where d_k is the number of silent polymorphic sites at which the k^{th} pair of sequences differ. Similarly, MCF is the maximum of x_i for all such fragments for all possible pairs of sequences.

Significance is assessed by carrying out a permutation test on the orders of the s silent polymorphic sites. Sites are permuted on the basis of their degeneracy in the amino acid code. So a column of data whose codon is twofold degenerate (i.e., two possible codons correspond to one amino acid) may only be replaced by another column of data which is also two-fold degenerate. A large number of such data sets are generated and a distribution of $SSCF$ or MCF under no recombination is found. This then allows the significance of the observed value to be assessed.

The test may be justified as follows: if there has been no recombination event since the most recent common ancestor of the sequences, then the distribution of bases at silent polymorphic sites should be determined by neutral mutation. Once the degeneracy of a site is determined, the distribution of bases should be independent of position. The permutation test preserves this dependency on the level of degeneracy at a position. Hence, differences in mutation rates should be separated from recombination events.

If a recombination event has occurred, then it will often result in an unusually long fragment. By the standard result that, subject to the constraint $\sum x_i = c$, $\sum x_i^2$ is minimised by placing equal values on the $x_i's$, a long fragment will tend to increase the value of $SSCF$.

Sawyer (1989) applies this test (or slightly modified versions) to several data sets, and found its performance satisfactory. However, it still does not address the problem of identifying the limits of recombination events. At this stage, Maynard Smith (1992) proposed the maximum chi-square test, which does find recombination breakpoints.

The maximum chi-square test can detect recombination and locate breakpoints in a segment of DNA provided the region is organised into two blocks, with one recombination breakpoint separating the two regions with different ancestral history. Maynard Smith (1992) considers two sequences, N base pairs long, which contain s polymorphic sites. An arbitrary cut is made after k sites, resulting in the sequences differing at r sites before the cut and $s - r$ sites after the cut. Obviously the expected numbers of polymorphic sites before and after the cut, assuming a random distribution, are $(k/N)s$ and [(N-k)/N]s. This allows the chi square statistic to be calculated. This process is repeated for all possible values of k until the cut which maximises the value of the chi-square statistic is found. This point is the location of the putative recombination event.

To test if this does, indeed, mark the limit of a recombination event, a permutation test may be used. For each randomised data set generated, the value of the maximal

chi-square statistic is found, and is used to form the distribution of the statistic under the null hypothesis that no recombination has occurred.

This test has been widely used in AIDS research (Robertson et al., 1995) and in other applications to detect recombination (e.g., finding recombination in strains of the Hepatitis B virus, Bollyky et al., 1996).

The maximum chi-square test does have several limitations. Firstly, the region must be in the two block structure described above. If, for example, a recombination event occurs in the middle of the sequences, such that the subsequences on either end have the same history while the central region has different ancestral relationships, then the maximum chi-square test may fail to find the recombination event. It is possible to split the data set up into smaller subsets and analyse each region separately, but this is tedious, and requires some prior knowledge about the locations of possible recombination events. Secondly, the maximum chi-square test considers only the polymorphic sites, so is not making the most efficient use of the information within the sequences. A further consequence is that a recombination breakpoint can only be located within the set of nucleotides lying between two polymorphic sites.

Maynard Smith (1992) describes an application to two sequences, Bollyky et al. (1996) describe an extension to 4 sequences. In practice data sets are considerably larger. It would be preferable to have a method which could be applied to larger numbers of sequences. This is not only beneficial from a practical viewpoint, but would also avoid the problems of multiple tests. If a larger data set has to be broken down into many subsets in order to look for recombination, it is quite likely that some of the results will be significant by chance alone. Multiple tests could be avoided if the researcher has some ideas, *a priori*, about possible recombinant strains and the corresponding parental lineages; this will also cut down on the amount of labour involved. Unfortunately, this will often not be the case.

A recent addition to methods based on polymorphic sites, the *homoplasy test*, was proposed by Maynard Smith and Smith (1998). This test determines if there is a significantly greater number of homoplasies in the most parsimonious tree estimated from the data set than would be expected under random substitution alone. The homoplasy test is suitable for sequences with low levels of divergence and Maynard Smith and Smith (1998) state that it should be considered as a complementary test to the maximum chi square test, which is suitable for sequences with greater levels of divergence.

Before describing this method, a homoplasy must first be defined. A homoplasy occurs when the same site mutates independently on different branches of a phylogenetic tree. An example is shown in Figure 3.2. The nucleotides at the tips of this tree are $GTTG$; these arose by two $T \longrightarrow G$ mutations on two different branches, as shown in the diagram.

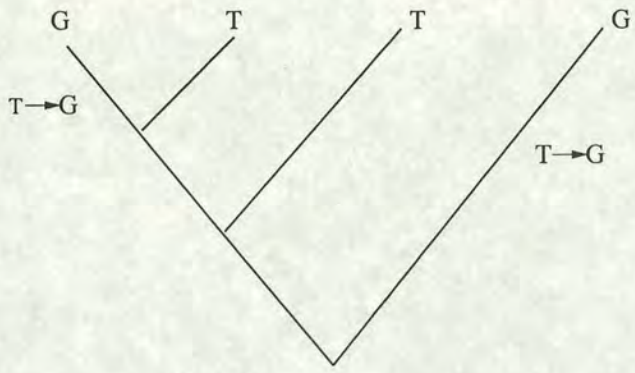


Figure 3.2: An example of a homoplasy.

To find the number of apparent homoplasies in a data set, the number of polymorphic sites and the most parsimonious tree (hereafter referred to as the MPT) are used. Let v be the number of polymorphic sites in the data set and let t be the number of steps or mutational changes in the MPT. Then the number of apparent homoplasies in a data set is given by $h = t - v$.

The expected number of homoplasies, $\widetilde{h}$, depends on the number of sites, N, in the data set. The larger the value of N, the smaller h should be (since it is less likely that the same site will mutate on more than one branch). Unfortunately the relationship between $\widetilde{h}$ and N is not simple; not all sites at risk of mutating are equally likely to change. Therefore, $\widetilde{h}$ is estimated by considering the effective number of sites, $N_E \leq N$. Given two identical genes affected by the same evolutionary forces, suppose that a random mutation occurs at one site in each of them. Let p_s be the probability that the same site changes in each of them. Then $N_E = 1/p_s$. Note that if there are N sites, all equally likely to change, the effective number of sites, N_E, will be the same as the actual number of sites, N, since $p_s = 1/N$.

Maynard Smith and Smith (1998) describe a method for estimating N_E using an outgroup to the set of sequences under analysis. They assume that the outgroup has been selected so that it satisfies the assumptions that it is subject to the same evolutionary forces as the data set, and that saturation in substitutions between the outgroup and the root of the data set (i.e., along the branch connecting the outgroup to the data set) has been achieved. If u is the number of changes along this branch, then $N_E = 2u$.

Once N_E has been found, a sampling distribution for the expected number of homoplasies may be found by simulation. Sites are selected, with replacement, from a set of N_E sites until v different sites have been chosen. If w is the number of selections required to achieve this, then $\breve{h} = w - v$ is the number of double hits and forms part of the empirical distribution of h under the hypothesis of no recombination. If the probability of observing $\geq h$ homoplasies is low, then the null hypothesis of no recombination may be rejected.

The homoplasy test has certain limitations. Firstly, since it considers only the polymorphic sites, it is losing a lot of the other information contained in the data set. In their examples using real data sets, Maynard Smith and Smith (1998) used only the synonymous changes as the third position, eliminating all others. Again information is being lost. In addition they assume that each site exists in only two states. Since the test has been developed for sequences with small amounts of change (1%–5%), this should not be a severe problem as transversional changes will be unlikely.

The homoplasy test does not estimate the locations of possible recombination events; it merely finds evidence for the presence of recombination in the data set as a whole. A simulation study conducted by Maynard Smith and Smith (1998) suggests that it requires a relatively large number of recombination events to have occurred before the test will have reasonable power. The simulation study was based on data sets of 16 species, with $N_E = 200$, which seems a reasonable value for data sets of closely related taxa. This is a potential problem, but requires further investigation.

The first three methods above consider the pattern of polymorphic sites within possible recombinant sequences, while the last examines the number of homoplasies within a data set. A different approach is described by Hein (1993). He detects recombination by considering changes in the most parsimonious topology along an alignment (thus, he is only using the informative sites within a multiple alignment). He starts by considering the possible new topologies that can arise from existing ones following one or more recombination events. This defines the set of topologies that must be considered given a particular starting topology.

The problem is then considered in terms of a graph. Each node (i, T) consists of the data at the i^{th} column of the alignment and a given topology, T. The node is assigned a weight, $w(i, T)$, the weight of position i given topology T. An edge connects two neighbouring nodes, i and $i-1$, and is assigned a weight $d(T, T')$, the recombinational distance between T (the topology at position i) and T' (the topology at position $i-1$). $W(i, T)$ is the weight of the most parsimonious history of the first i positions, given that the topology at position i is T.

The most parsimonious history of the sequences will be the path of lowest weight from node 1 to node N, where N is the sequence length, the weight being found by summing the weights of the nodes and the edges. This is given by $W(N, T)$, found by the following recursion:

$$\begin{aligned} W(1,T) &= w(1,T) && (i=1) \\ W(i,T) &= \min_{T'} \{W(i-1,T') + d(T,T') + w(i,T)\} && (i>1) \end{aligned} \qquad (3.1)$$

Thus, the sequence will start in a particular topology and will only change topologies when it becomes worthwhile to do so [sufficiently low values of $w(i, T)$]. A sensible

choice of values of $d(T, T')$ will prevent too frequent changes in topology (e.g., after a couple of nucleotides).

The dynamic programming algorithm described by Hein (1993) does yield exact results, and is relatively fast for small data sets. However, it is impractical for more than five or six sequences. Since it is useful to have a method which may be applied to larger data sets, Hein (1993) describes a heuristic version of this algorithm. While applicable to most practical problems, it no longer guarantees that it will find the cheapest path from node 1 to node N.

The heuristic algorithm makes some basic assumptions. Firstly, it is assumed that only one recombination event happens between each node (nucleotide). Therefore, all topologies which are separated by two or more recombination events from topology T may be discarded when $W(i, T)$ is being calculated. It is also assumed that the correct topology is known at some point in the sequence (e.g., at the first nucleotide). The algorithm then starts with this topology and scans topologies which are one recombinational step away. This continues until a new topology is selected; the algorithm then starts to scan trees in the neighbourhood of this topology. Of course, in practice, the correct tree will not be known for any node. The topology based on the entire sequence may be used as an approximation (this is more likely to be correct than a random tree, particularly if recombination is a rare event). As a check of this starting topology, the algorithm may be run in reverse, starting at node N, and the results may be compared.

Hein (1993) notes that the parsimony algorithm has been criticised (see **2.4**). Since his proposed algorithm for detecting recombination is based on the parsimony principle, it is likely that it will also suffer from these same problems. However, as a first approach to tackling the difficult problem of detecting recombination, it can be justified. The advantage of the parsimony criterion is that it leads to a well-defined minimisation problem which can be solved as outlined above.

It may be reasonable to extend the idea behind this algorithm to the more statistically sound distance and likelihood methods and use these as a basis for detecting recombination. Indeed a similar concept is used in Chapter 5, where the theory of Hidden Markov models is used to develop a Bayesian approach to the detection of recombination.

3.3 Approaches using the non-parametric bootstrap

Other authors considered the use of alternative tools for detecting recombination. For example, Salminen et al. (1995) developed a procedure which they term *bootscanning* for detecting recombination in strains of HIV-1. The essentials of their method are as follows. A database is maintained of representative nonrecombinant sequences of established genotypes of HIV-1. To test a sequence for recombination, a data set is

built up consisting of that sequence, and strains of the established genotypes from the database. A moving window (length 200-500 bp) slides along the alignment, creating overlapping segments, on each of which a phylogenetic bootstrap analysis is carried out. If a DNA sequence has been subject to recombination events in the past, then different segments of this sequence will cluster with different genotypes. Once possible recombination events have been detected in a sequence, that sequence and the parental strains (and an outgroup) may be reanalysed, and the location of the recombination breakpoints pinpointed more accurately. This is done by noting where high bootstrap support for clustering with one of the parental genotypes suddenly changes to high bootstrap support for clustering with another.

This method is based on a good premise, and since detecting recombination is of vital importance in AIDS research, it could play an useful role. However, it is limited by the fact that it requires a database of established genotypes of the organism in question. Thus, a researcher must assemble such a database, if one does not already exist, which is time consuming. Another drawback is the computational burden involved in implementing large numbers of bootstrap analyses.

A different approach using the bootstrap was suggested by Lawrence and Hartl (1992). They consider two data sets of N sequences: a reference data set and a test data set. In the absence of recombination the two data sets should have the same phylogenetic history.

In order to compare the two data sets, Lawrence and Hartl (1992) firstly compute the pairwise percentage similarity (number of identical sites / total number of sites $\times$ 100) for all pairs in each of the data sets. Two matrices are formed which contain the similarity measures. In order to make the two matrices commensurate, the magnitude of the relationships in each row of the matrices are ranked (i.e., the first row contains the percentage similarities between the first sequence and all others; it is these that are ranked). The Spearman rank correlation statistic is then calculated for each row in the two matrices as follows: if ρ_{1i} and ρ_{2i} are the i^{th} entries in a particular row in the first and second rank matrices respectively, then the Spearman rank correlation statistic is given by

$$S = 1 - \frac{6 \sum_{i=1}^{N} (\rho_{1i} - \rho_{2i})^2}{N^3 - N} . \tag{3.2}$$

Note that $S = 1$ indicates perfect positive correlation, while $S = -1$ indicates perfect negative correlation.

Once the Spearman rank correlation statistic has been obtained for each row, an overall similarity coefficient may be found by averaging the row statistics. Coefficients less that 1 suggest that there is some discrepancy between the test and reference data sets.

A bootstrap analysis is used to assess the significance of this similarity coefficient. As in the standard applications of bootstrapping in phylogeny, the columns of the reference data set are sampled with replacement to form k new data sets. Each of these is compared to the reference data set in the manner described above, and a similarity coefficient is obtained for each. The distribution of these coefficients which results from this procedure shows the variation expected due to random stochastic error, and may be used to assess if the observed value of the similarity coefficient is significantly smaller than expected if no recombination has occurred.

Lawrence and Hartl (1992) point out that if there is only one recombinant sequence in a data set, the removal of this sequence would mean that the reference and the test data sets no longer differ significantly. Therefore, the analysis could be repeated with one sequence omitted each time in an attempt to identify the recombinant. It must be noted, however, that significance levels must be altered to avoid spurious results due to the problems of multiple comparisons.

This method does not identify recombination breakpoints in a sequence, which is a limitation. As it uses the entire sequence length, it may lack the power to find short recombination events relative to the entire sequence length. It is also possible that for large data sets (i.e., large numbers of sequences) with only one recombination event, the information on the discrepancy may be swamped by the good matching of the other sequences. A way around this would be to use subsets of the larger data set, but again this requires that the researcher has some prior knowledge about possible recombinant sequences and their parental strains.

3.4 Likelihood-based procedures for detecting recombination

Likelihood methods generally make very efficient use of the information contained in a data set; therefore it seems obvious to tackle the problem of detecting recombination using likelihood in some guise.

Huelsenbeck and Bull (1996) considered the simpler problem of detecting conflicting phylogenetic signal from data sets containing different parts of the genome. They developed a method which can detect sources of heterogeneity, in general, between data sets, although their specific application looks at changes in the branching pattern, and thus should detect recombination.

Their procedure uses a likelihood ratio test to evaluate the null hypothesis that differences in phylogenetic estimates are a result of random stochastic error, rather than heterogeneities in the data sets. The alternative hypothesis allows the different data sets to have different phylogenies.

Suppose the model parameters of the i^{th} data set are the ordered pair $\boldsymbol{\theta}_i = (T_i, \boldsymbol{\phi}_i)$

where T_i represents the topology and ϕ_i denotes the other phylogenetic parameters (e.g., branch lengths, transition-transversion ratio etc). These quantities are estimated from each of the data sets, yielding the set of estimates $\omega = \{\theta_1, \ldots, \theta_N\} \in \Omega$ for all N data sets. To specifically test for changes in the branching pattern, calculate the likelihood, L_0, under the null hypothesis:

$$L_0 = \max\{L(\omega)\}\big|_{\omega\in\Omega; T_1=T_2=\ldots T_N}.$$

Note that, while the topologies are constrained to be equal, the other phylogenetic parameters may vary from one data set to the next.

This is compared to the likelihood, L_1, under the alternative hypothesis which allows the topology as well as the other parameters to vary across the data sets:

$$L_1 = \max\{L(\omega)\}\big|_{\omega\in\Omega}.$$

Then the likelihood ratio statistic is given by

$$\delta = 2[\ln L_1 - \ln L_0]. \tag{3.3}$$

As mentioned in **2.8**, phylogenetic likelihood ratios often do not have asymptotic χ^2 distributions. Therefore, it is necessary to find the null distribution of δ using Monte Carlo simulation or parametric bootstrapping. Since the true values are unknown, the one underlying topology and other parameter values must be estimated from the data. A large number of data sets may then be simulated and the distribution of δ under the hypothesis of no heterogeneity in branching pattern from one data set to another may be found.

Clearly this test would be useful if potential recombination breakpoints are known. Since it is based on likelihoods, it makes efficient use of the data in the various data sets. Therefore, it should have greater power than tests based merely on the polymorphic or informative sites. If some initial scanning method is used to detect possible recombination breakpoints, however (such as the algorithm using the Dss statistic described in Chapter 4), the different subsets will be selected on the basis of maximal difference, so a bias will be introduced into the likelihood ratio test. Hence, it will be necessary to increase the significance level; a sufficiently stringent level should offset this bias.

A different approach using likelihoods was proposed by Grassly and Holmes (1997). They consider the fundamental problem of detecting recombination in a data set where there is no prior knowledge about whether recombination has even occurred, or the sequences involved in the event. Intuitively their idea is quite simple: consider the maximum likelihood phylogeny for the entire data set, and look at the values of the likelihood at each site. If a recombination event has occurred, then this phylogeny will be a poor fit to the data in that region, and should be reflected by lower site likelihood values.

In more detail, their approach proceeds as follows: once the site likelihoods, according to the maximum likelihood phylogeny for the entire data set have been found, a sliding window of varying length (from 5 base pairs to half the sequence length) moves along the sequence. In each window of length s, from sp to $sp + s - 1$, the following quantity is calculated:

$$Q = \frac{\sum_{i=sp}^{sp+s-1} \ln L_i}{s} \Bigg/ \frac{\sum_{i=1}^{sp-1} \ln L_i + \sum_{i=sp+s}^{N} \ln L_i}{N - s}, \tag{3.4}$$

where L_i is the likelihood at each site and N is the total sequence length. Essentially, Q is finding the ratio of the average log likelihood in a region compared to that in the rest of the sequence. High values of Q correspond to regions of low likelihood and suggest heterogeneity in the data.

It is necessary to assess the significance of the values of Q before any conclusions may be drawn about the data set. Thus, a distribution of the maximal Q values under the null hypothesis of no recombination, or other heterogeneities in the data must be found. This was initially done using parametric bootstrapping, the largest value of Q for each window size from each simulated data set being recorded. However, Grassly and Holmes (1997) noted that this distribution appeared to be normal (confirmed by a Kolmogorov-Smirnoff test), so they conclude that normal distribution theory may be used to assess the significance of Q.

Simulation results and examples using real data suggest that this method performs well. Nonetheless, it is not without its problems. Firstly, it may not be able to distinguish between recombination events and rate variation. Since the likelihood values at each site are calculated according to a single maximum likelihood phylogeny with fixed branch lengths, regions of the alignment with different branch lengths due to rate variation may have lower likelihood values which could be significant. One way of dealing with this problem would be to incorporate rate variation into the model used to find the maximum likelihood tree; for sequences with unknown regions of variation in mutation rates, the Hidden Markov model approach for rate variation, as described by Felsenstein and Churchill (1996), could be used.

Another problem stems from the fact that one tree is used to calculate the site likelihoods. If the recombinant regions are quite large relative to the entire data set, then the maximum likelihood phylogeny will be some type of average of the different trees along the sequence. Therefore, the site likelihood values will not be differentiated by as much as if the maximum likelihood tree was exactly correct for parts of the sequence and not for others. This may cause the test to lose power. An approach considering local trees (i.e., trees estimated on subsets of the entire alignment) may be more powerful; this is taken into consideration in both of the methods described in Chapters 4 and 5.

3.5 Split decomposition

Split decomposition, developed in the phylogeny context by Bandelt and Dress (1992), is a non-approximate method which allows for conflicting groupings of sequences. It will find possible phylogenetic relationships even if, for other phylogenetic methods, the signal is overridden by other possible groupings. Therefore, it should be able to detect (and display) the conflicting information that usually arises when a recombination event has occurred.

Bandelt and Dress (1992) suggest using split decomposition with distance matrices. Recall from **2.7** that phylogenetic tree estimation methods which use distances usually require that the distances are additive (i.e., they satisfy the four point condition: if taxa 1 and 2 are separated by an edge from taxa 3 and 4 then $d_{12} + d_{34}$ is smaller than $d_{13} + d_{24} = d_{14} + d_{23}$). In practice, distance estimates are rarely additive; a more relaxed approach would have $d_{12} + d_{34} < \max\{d_{13} + d_{24}, d_{14} + d_{23}\}$. This suggests a criterion for finding splits in a data set.

A data set may be partitioned into two sets $\mathcal{A}$ and $\mathcal{B}$ (called a *split*) if, for any $i, j \in \mathcal{A}$ and $k, l \in \mathcal{B}$

$$d_{ij} + d_{kl} < \max\{d_{ik} + d_{jl}, d_{il} + d_{jk}\}.$$

Bandelt and Dress (1992) refer to this as a *d-split*. Every d-split carries a weight called the *isolation index*, which is given by

$$\alpha_{\mathcal{A},\mathcal{B}} = \frac{1}{2} \min_{\substack{i,j \in \mathcal{A} \\ k,l \in \mathcal{B}}} \left[\max\{d_{ij} + d_{kl}, d_{il} + d_{jk}, d_{ik} + d_{jl}\} - d_{ij} - d_{kl}\right]$$

From this definition it is seen that all partitions which are not d-splits have an isolation index of zero. Also, for a tree with additive data, the isolation index of $\mathcal{A}, \mathcal{B}$ is the length of the edge whose removal results in the two components $\mathcal{A}$ and $\mathcal{B}$.

Every d-split, $\mathcal{A}$, $\mathcal{B}$, yields a *split metric*, $\delta_{\mathcal{A},\mathcal{B}}$, which assigns a distance of one to taxa i, j if $i \in \mathcal{A}$ and $j \in \mathcal{B}$ or vice versa and zero otherwise. The sum, d^1, of all split metrics, weighted by their isolation indices, approximates the total distance d from below by

$$d = d^0 + \sum_{\text{splits } \mathcal{A},\mathcal{B}} \alpha_{\mathcal{A},\mathcal{B}} \delta_{\mathcal{A},\mathcal{B}}$$

where the last term is, obviously, d^1. The residue, d^0, does not contain any further splits with a positive isolation index. For real data, d^0 is usually non-zero, so to measure the efficiency of the split decomposition in describing the relationships within the data, the matrices d and d^1 are compared, yielding

$$\rho := \left(\sum_{\text{taxa } i,j} d^1_{ij} / \sum_{\text{taxa } ij} d_{ij} \right) \times 100\%,$$

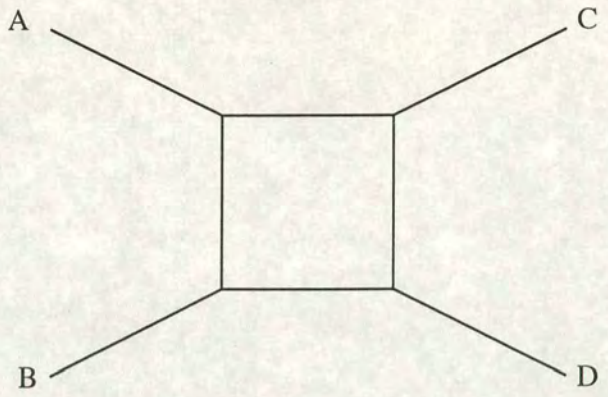


Figure 3.3: an example of a network. There is support for both of the clusters AB and AC.

the splittable percentage. Clearly, the higher the value of ρ, the better the data are explained by the d-splits.

Since split decomposition allows for conflicting relationships, the results are displayed as a network, an example of this for four taxa is shown in Figure 3.3. Here the splits AB/CD and AC/BD are both valid for the data (e.g., because of a recombination event). Note that it is possible to estimate trees from a split decomposition: an optimal set (under some criterion) of splits which are pairwise compatible are chosen (two splits $\mathcal{A}, \mathcal{B}$ and $\mathcal{C}, \mathcal{D}$ are pairwise compatible if there exists $\mathcal{J} \in \{\mathcal{A}, \mathcal{B}\}$ and $\mathcal{K} \in \{\mathcal{C}, \mathcal{D}\}$ such that $\mathcal{J} \cap \mathcal{K} = \emptyset$). It is possible, however, that some furcations may be left unresolved.

While split decomposition does provide a way of finding and displaying conflicting relationships within data, it is not the most powerful tool for detecting recombination. Bandelt and Dress (1992) observe that it is not obvious how to discriminate between random or systematic error in the data set, and convergent evolution or recombination events. This is due to the fact that their method uses distances rather than character state data. Therefore, if it is important to find a recombination event, another method may be more appropriate.

Chapter 4

A Graphical Method for Detecting Recombination in Phylogenetic Data Sets

A graphical method, using a statistic termed *Dss*, for initially scanning DNA data sets for evidence of recombination is described in this chapter. It is applicable to large data sets and does not require a large amount of computational time. Therefore, it should compliment the available tests for recombination described in Chapter 3, many of which are only applicable to small data sets or carry a large computational load.

This chapter opens by describing some of the aims and the motivation which led to the development of the *Dss* statistic. The *Dss* statistic is defined and its expected behaviour discussed. The method is then evaluated by simulation, and by application to some real data sets. Finally, possible improvements of this method are discussed. Note that much of the work in this chapter has been previously reported in McGuire et al. (1997), while the computer package written to implement the necessary calculations is described in McGuire and Wright (1998).

4.1 Motivation

When planning this work, there were several objectives which were thought to be important. Firstly, since most data sets are large, the method should be applicable to more than a handful of sequences. At the time of planning, the maximum chi-square test (Maynard Smith, 1992) was the most frequently used test for recombination; this may be only applied to four sequences at most (see **3.2**).

Suppose the branching patterns of trees estimated from subsets of the alignment are considered. Changes in the topologies of these 'local' trees along the alignments suggest that recombination may have occurred in the past, and tests using local trees should be more powerful than those based on one global tree estimated from the entire alignment (see the discussion on PLATO in **3.4**, Grassly and Holmes, 1997).

Many methods for detecting recombination make use only of the polymorphic or informative sites (see **3.2**). Distance and likelihood methods for phylogenetic inference make more efficient use of the data. For reasons of computational speed, it was decided to use some of the distance methods rather than likelihood methods.

Finally, a relatively quick method for scanning a DNA alignment for recombination prior to a full phylogenetic analysis would be of use to a biologist. Thus, emphasis was placed on speed, rather than developing a comprehensive method for statistically testing for the presence of recombination. While it is possible to do some approximate statistical tests (discussed at the end of this chapter), further analysis using some of the methods discussed in Chapter 3 is recommended.

4.2 Definition of the *Dss* statistic

Consider a data set of n aligned DNA sequences, each of length N, and a window of length $2l$ base pairs which moves along the sequence from beginning to end in increments of m base pairs, $m << 2l$. The number of overlapping windows that results is then

$$W = \frac{\widetilde{N} - 2l}{m} + 1 \tag{4.1}$$

where

$$\widetilde{N} = \begin{cases} N & \text{if } m \text{ is a factor of } N, \\ \max\{k; m|k, k \in \mathbb{N}, k < N\} & \text{otherwise.} \end{cases}$$

Each window is split into two equal parts, each of length l. On the first half of the window, a distance matrix is calculated according to some Markov model of nucleotide substitution (see **2.7.1**). A phylogenetic tree is estimated on the first half of this window using the least squares method (Cavalli-Sforza and Edwards, 1967; Fitch and Margoliash, 1967, see **2.7.5**). This optimal tree has a *sum of squares* value associated with it; this is recorded as SSa^F. Note that this value should be quite low since the selected tree is optimal according to the least squares criterion.

A distance matrix is then calculated for the second half of the window using the same model of substitution as before. The topology estimated from the first half of this window is fitted to this second distance matrix, again using least squares. Its associated sum of squares value is also recorded as SSb^F. Then the Difference in the Sum of Squares statistic, going Forward, is defined as

$$Dss^F = SSb^F - SSa^F. \tag{4.2}$$

This statistic is calculated for all possible windows, with index i ($i = 1, \ldots, W$), along the sequence, yielding a set of values, $\{Dss_i^F\}$.

The process is then repeated in the backward direction (i.e., the first window is at the end of the sequence, and the windows slide backwards, moving in steps of m base pairs each time). Again least squares is used to estimate a tree from the first half of the window, yielding SSa^B. This topology is then fitted to the distance matrix from the second half of the window, producing SSb^B. DSS_i^B is calculated as $SSb^B - SSa^B$ for each window i, $i = W, W-1, \ldots, 1$. Finally the overall Dss statistic is defined as

$$Dss_i = \max\{Dss_i^F, Dss_i^B\}, \tag{4.3}$$

yielding the set of values $\{Dss_i\}$. These may be plotted against the centre of each corresponding window, and the resulting graph used to scan for recombination. The reason for this particular definition of Dss is explained below.

4.3 Expected behaviour of the *Dss* statistic

Various factors influence the behaviour of the Dss statistic, and its constituent parts, the $\{Dss_i^F\}$ and $\{Dss_i^B\}$. Firstly, the effect of recombination on Dss is examined so that it is possible to recognise putative recombination breakpoints. Dss is affected by other things such as tree length, rate variation, branch length and window size. These are also detailed below.

4.3.1 Recombination

Consider the value of the Dss statistic within a particular window. Suppose, firstly, that no recombination has occurred within this window. Then all sites will have the same underlying topology. In particular, the branching pattern on the first half of the window is expected to be the same as that on the second half (any differences should be small, and be the result of stochastic error). This has the consequence that the optimal topology for the first half of the window should be very close to, if not the optimal branching pattern for the second half. Thus, SSb^F or SSb^B will be of small magnitude, and hence Dss will be close to zero. Therefore, regions in an alignment containing no recombinant sequences should correspond to low values of Dss.

Suppose now that a recombination event, which changes the branching order, has occurred within the window, with the breakpoint at the centre of the window. Then the topology on the first half will be different to that for the second half of the window. Hence, the optimal topology for the first half of the window will be a poor fit to the distance matrix from the second half of the data, and this will be reflected in a high value of SSb^F or SSb^B, which in turn leads to a high value of Dss.

Most windows containing a recombination event will not have the breakpoint located at the centre of the window. Suppose that the first half of the window is all topology 1 say, while the second half of the window has a breakpoint located within it; initially

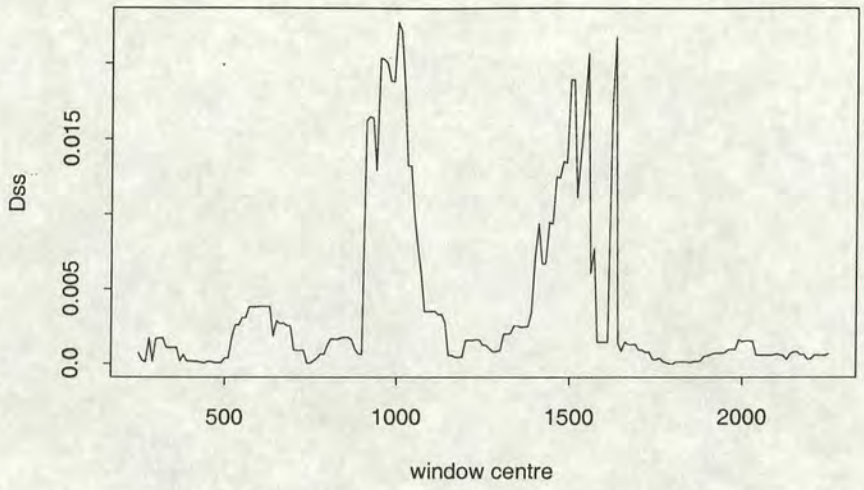


Figure 4.1: An example of the graphical output for a 10 sequence simulated data set, with an easily-detectable recombination event with breakpoints at 1000 and 1500 bp.

the topology is that of the first half, but then it changes to topology 2. The optimal tree for the second half of the window will be some type of average between these two topologies (depending on the relative strengths of their signals), so will still be different to the topology on the first half, though not to the same extent as if the breakpoint was in the centre of the window. Thus, the Dss value should still be higher than if there had been no recombination, but not as high as the Dss value when the recombination breakpoint is in the centre of the window. Pooling all this information, it is concluded that a recombination breakpoint in an alignment should be marked by a peak in the Dss values, with the highest value being the estimate of the location of the breakpoint.

An example of the output from this algorithm is shown in Figure 4.1. The data set is simulated along the phylogeny shown in Figure 4.4, using the Jukes-Cantor model of nucleotide substitution. A recombination event with two breakpoints at 1000 and 1500 bp is simulated, and is an ET event at the first depth (see **4.4.1** for full details; essentially this is a recombination event which should be easily detected). The Dss values were calculated using a window of 500 bp which is moved along in increments of 10 bp. The distances were calculated using the Jukes-Cantor model; unweighted least squares was used to find the sums of squares in each window.

The output is easy to interpret. There are two relatively large peaks, centred around 1000 and 1500 bp, suggesting that limits of a recombination event occur in these regions. Elsewhere, the values tend to be low; fluctuations are due to random noise. Note that because the Dss values are positively correlated, a pattern of peaks and troughs will be observed in the non-recombinant regions. However, peaks due to recombination tend to be larger, both in height and sometimes in width, as may be seen from Figure 4.1.

4.3.2 The effect of tree length

It was noted above that the definition of the Dss statistic appears to be somewhat convoluted. The question must be asked as to whether or not the $\{Dss_i^F\}$ or the $\{Dss_i^B\}$, on their own, contain the information needed. In general the answer is no.

Initial work on the properties of the Dss statistic with data sets containing two recombination breakpoints found that the reduction in size, or even the complete absence of one of the peaks corresponding to a breakpoint appeared to be a problem for the $\{Dss_i^F\}$ and the $\{Dss_i^B\}$. This was even the case for data sets with very recent recombination events between two distantly related taxa, an event which should be straightforward to detect. To illustrate this, a data set was simulated according to the tree shown in Figure 4.4, with an easily detectable recombination event (the ET type at the first depth; see Figure 4.4 for details). The recombinant region has breakpoints located at 1000 and 1500 bp. Figure 4.2 shows the plots of the resulting $\{Dss_i^F\}$ and $\{Dss_i^B\}$ values against the corresponding windows.

Both peaks are present in the graph of the $\{Dss_i^F\}$ although the second peak is somewhat larger. The problem is clearly illustrated in the middle graph which contains the $\{Dss_i^B\}$. In this graph the second peak completely disappears. Defining Dss as the maximum of the forward and backward values in each window appears reasonable; in the bottom graph showing the Dss values plotted, both peaks corresponding to the limits of the recombination event are present. The question still remains, however, as to what artifact in the data is causing this suppression of peaks in the forward and backward values.

Upon further investigation, it appeared that changes in the tree length were at the root of this problem. A recombination event will often change the total length of a tree (sum of all the branch lengths). As a result, a recombination breakpoint may mark a transition from a longer tree to a shorter one and vice versa. Dss is dependent on the length of the tree (see equation 4.4 below); longer trees tend to have higher values of the sum of squares. If there is a recombination event in a window, such that the first half has one tree, while the second half has a different topology which is also a longer tree, then the value of SSb will be considerably larger than SSa, not only due to the discrepancy between topologies, but also because of the greater tree length. On the other hand, if the tree in the second half of the window is shorter, then the increased value of SSb due to the recombination event is offset, to some extent, by the reduction in the sum of squares due to the shorter tree. This will lower a peak in the Dss values due to recombination, and in some cases may even cause it to disappear. By finding the Dss values going both forwards and backwards along the sequence, and then taking the largest one, only those values which are inflated by transitions to longer trees should be selected. Therefore, all peaks due to recombination should be found.

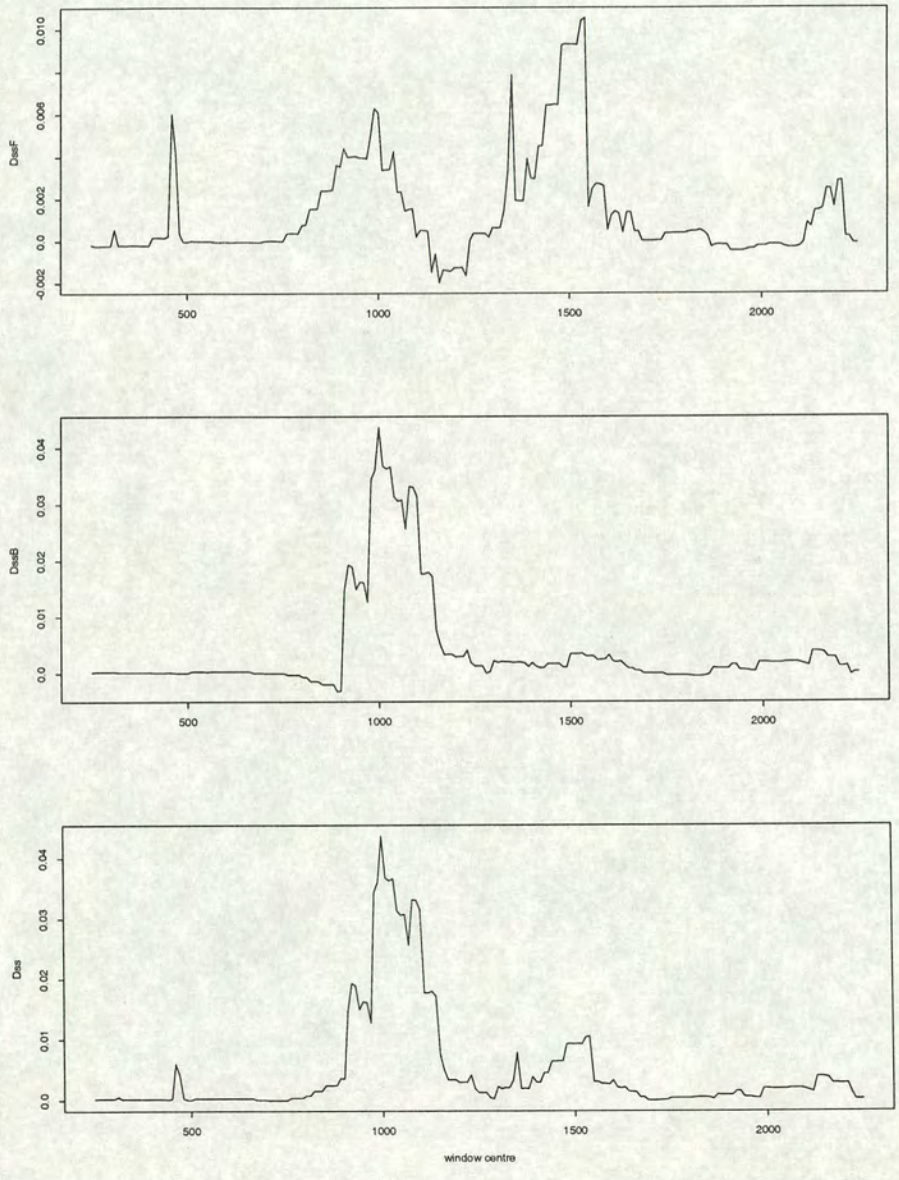


Figure 4.2: The $\{Dss_i^F\}$ (top graph) and the $\{Dss_i^B\}$ (middle graph) plotted against each window centre. The bottom graph shows the corresponding Dss values for each window. Note the different scales between the first, and the second and third graphs.

4.3.3 Weighted v unweighted least squares

In the definition of the Dss statistic above, it was not indicated whether unweighted (Cavalli-Sforza and Edwards, 1967) or weighted (Fitch and Margoliash, 1967) least squares should be used. The sum of squares criterion (also see equation 2.44) is

$$SS = \sum_{i=1}^{T-1} \sum_{j=i+1}^{T} d_{ij}^{-P} (d_{ij} - p_{ij})^2 \tag{4.4}$$

where SS is the sum of squares [the error in fitting the distance estimates to the tree, E in (2.44)];

T is the number of species;

d_{ij} is the estimate of the pairwise distance between sequences i and j;

p_{ij} is the predicted distance between sequences i and j, from the tree;

d_{ij}^{-P} is the weight applied to the branch lengths between sequences i and j, where $P \in \mathbb{R}$, $P \geq 0$. P is often referred to as the *power*.

Weighted least squares (i.e., an appropriate value of $P > 0$) should standardise the sum of squares, and therefore Dss for varying branch lengths along the alignment. An example of where this might come in useful is if there is a region within a set of DNA sequences with higher nucleotide substitution rates, resulting in longer branch lengths in that part of the sequences. Even if no recombination has occurred in the sequences, peaks in the $\{Dss_i\}$, found using unweighted least squares, are quite likely to occur, marking the boundary of this region of increased variation. This is caused by the effect of the longer tree length on Dss, as explained above. Since the aim of this work is to detect recombination, allowing for rate variation by using weighted least squares sounds reasonable to reduce confounding between rate variation and recombination.

The two graphs on the left-hand side of Figure 4.3 show a case where using weighted least squares to allow for rate variation proves beneficial. The data used are simulated according to the topology in Figure 4.4 with the branch lengths in the same proportions, though a different basic length is used, the basic branch length in Figure 4.4 being 0.1. No recombination event occurs, but the subsequence from 1000 bp to 1500 bp evolves three times faster than the rest of the sequence. In the slower-evolving parts of the alignment (1–1000 bp and 1501–2500 bp), the basic branch length is 0.04, while in the more diverged region, the basic branch length is 0.12. The Dss values calculated using unweighted least squares have peaks around 1000 bp and 1500 bp, marking the limits of the faster evolving region. To try and standardise for varying branch lengths along the alignment, weighted least squares was then used to calculate the values in the lower left-hand graph in Figure 4.3. A sensible choice for P in (4.4) is 2 (Fitch and Margoliash, 1967). Weighted least squares does appear to reduce the effect of the rate variation – the first peak disappears and the second one is no longer as pronounced.

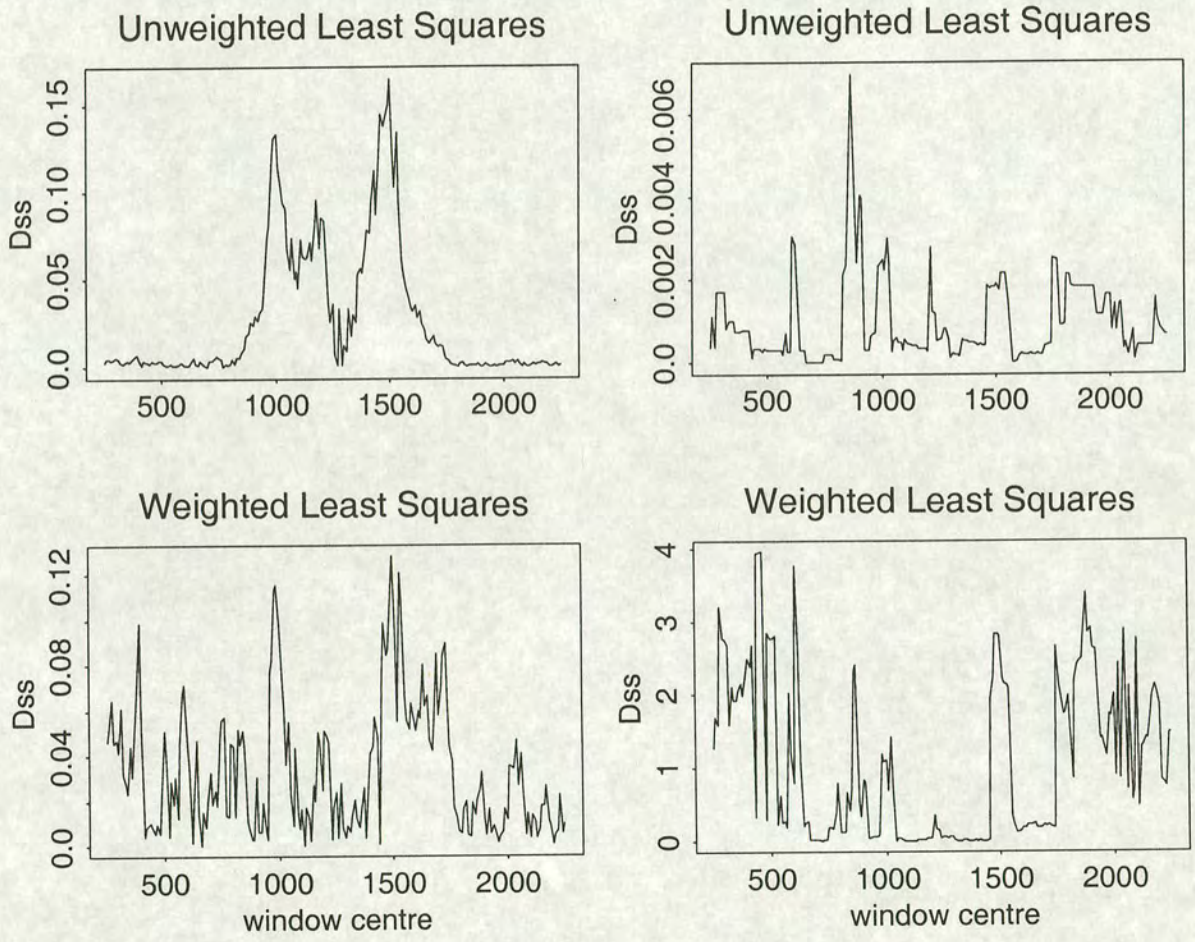


Figure 4.3: *Dss* calculated for a data set containing no recombination events, but with substitution rate variation. Unweighted (top graphs) and weighted (bottom graphs, power=2) least squares are used. The left graphs show an example where weighted least squares may be used; the right graphs show an example where it should not be used, due to the short branches problem.

Unfortunately, using weighted least squares does not always lead to sensible results from this algorithm. Suppose that some of the pairwise distances between the taxa in the data set are small (i.e., close to zero). Using a power of two means that the denominators of these terms in the sum of squares will be very small (the result of squaring a number less than one). Small differences (due to sampling error) in the estimation of short branch lengths in adjacent regions could result in a relatively large effect on the *Dss* statistic. This is the effect seen in the right-hand graphs in Figure 4.3. The same topology is used as before but this time the basic branch length in the slow region is 0.005 and in the fast region is 0.015. The graph of the *Dss* values calculated using weighted least squares contains a certain number of sudden, large fluctuations in the values of *Dss*, and thus is difficult to interpret. For this particular data set, the presence of rate variation does not lead to clear peaks in the plot of the *Dss* values using unweighted least squares (while there is a high peak before 1000 bp, it is not particularly wide) so it is not really necessary to use weighted least squares to take account of rate variation. For real data sets, the presence of rate variation may be suspected, but the short branches problem may mean that unweighted least squares must be used to calculate *Dss*. Thus, any peak must be checked using other tests to see if it is, in fact a recombination breakpoint, rather than the limit of a more diverged region.

One further comment is that the power may take on any positive real value. Hence, it could be varied continuously between zero and two, say, thus allowing perhaps a trade-off between accounting for rate variation (should improve with power) and the short branches effect (worsens with increasing power). It might be possible to fine-tune the choice of power for a particular data set to yield a *Dss* statistic which accounts, to the best of its ability, for rate variation, while avoiding the short branches affect. This is a point which requires further investigation.

4.3.4 Window size and increment

The *Dss* values are also affected by the choice of window size. Since a distance matrix and a phylogenetic tree are estimated from each half of the window, it is important that the window is long enough to contain enough information for this purpose. If the window is too short, then the pairwise distance estimates and thus the values of the *Dss* statistic will be very noisy and this will often override any signal from a recombination event in the data. Initial work suggested that windows of 200bp were too short, but sizes of 400 or 500 bp or greater were more useful. Another important point is that the more complicated the model of substitution used, the longer the window should be since the greater number of parameters introduces a higher level of variability into the distance estimates.

However, it is not necessarily the case that the bigger the window size the better. If a very short recombination event has occurred, then it might be difficult for *Dss* based on a large window size to detect this event, whereas the statistic calculated using a smaller window might find it. In addition, it is important to have enough windows to be able to examine the behaviour along the alignment. Hence, the choice of increment is important since that also plays a part in determining the number of windows.

Some of the properties of the *Dss* statistic have been detailed above. Its expected behaviour in the presence of recombination has been discussed. However, it is still necessary to validate that it does, indeed, successfully locate recombination breakpoints. Below, details of a simulation study which was carried out to investigate the performance of *Dss* are given.

4.4 A simulation study to investigate the performance of *Dss*

The details and results of a simulation study carried out to assess the performance of *Dss* are given below. Firstly, the method used to generate the data and simulate recombination events is discussed. A heuristic way of measuring the difficulty of detecting a recombination event is given, followed by the results of the simulation study.

4.4.1 Data simulation

Data sets were simulated under a variety of conditions, in order to evaluate the effectiveness of the *Dss* statistic. The nonrecombinant phylogeny used is that shown in Figure 4.4. Each data set was simulated using the Jukes-Cantor model of nucleotide substitution (Jukes and Cantor, 1969, see **2.5**). The sequence length was 2500 bp. Each recombination event involved the nucleotides between positions 1000 and 1500 bp in the alignment, and involved the substitution of that region of DNA in one sequence for the corresponding region in another.

Several types of recombination event were simulated. These events can be broken down into two main subgroups: a *half-tree* (HT) event and an *entire-tree* (ET) event. HT recombination events involve sequences in the top half of the tree (those along the short dashed line in Figure 4.4). HT recombination events occur at three different depths in the tree as marked in the diagram; these depths are half-way along the branch in question. When a recombination event occurs at depth i, say, the sequences are simulated along the phylogeny in Figure 4.4 as far as that particular depth. At this point the 1000-1500 bp region from the lower positioned sequence in the diagram replaces the corresponding region in the higher sequence (this means that these two sequences [and any descendants, if applicable] will cluster together). The sequences are

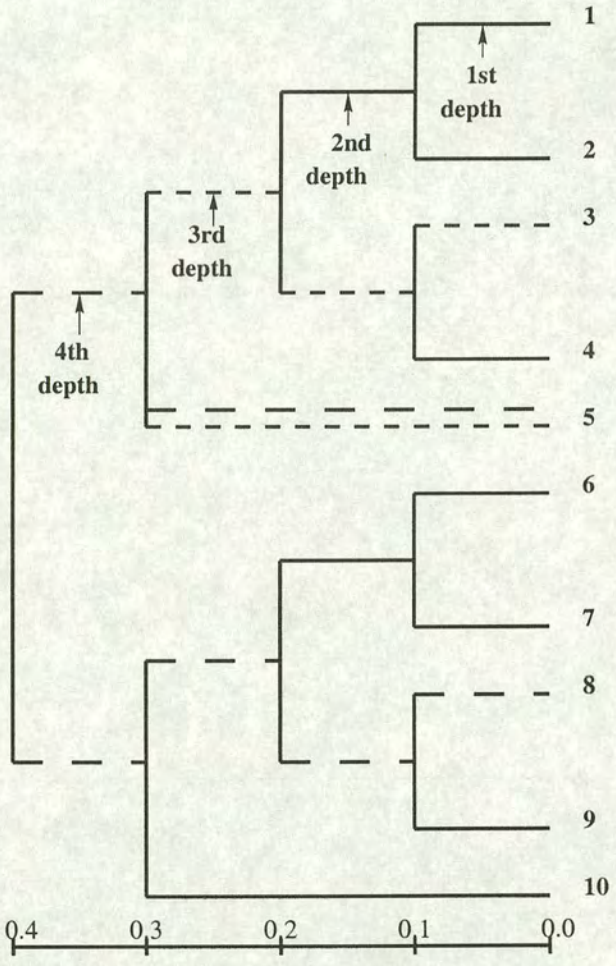


Figure 4.4: The tree used to simulate the data sets and the branch scale. - - -. HT recombination event; — — —, ET recombination event. Recombination occurs between the branches marked by the short/long dashed lines at the three/four different depths in the tree, as shown. The event happens halfway along the two branches in question: for example, the ET recombination event at the fourth depth occurs when the two sequences involved have diverged by 0.05 substitutions per position.

then evolved along the remaining part of the phylogeny. An ET recombination event is simulated in a similar manner to a HT event, the only difference being that an ET event involves sequences across the entire tree (see the long dashed line in Figure 4.4).

In total, there are seven different types of recombination event; for each event one hundred data sets were simulated. In order to evaluate the Dss statistic, it was necessary to have a measure of its values in the absence of recombination. Therefore, two hundred data sets were generated according to the phylogeny shown in Figure 4.4. The Jukes-Cantor model for nucleotide substitutions was used, as before.

To calculate the Dss values, a window of 500 bp, moving in increments of 10 bp was used. This yielded 201 Dss values, a reasonable number of values with the computational time being kept at a sensible level (each run required less than 10 minutes CPU time on the MRC HGMP Research Centre computing facilities, a Sun Ultra Enterprise, 20 $\times$ 167 MHz processors and 1 Gbyte of memory; Rysavy et al., 1992). Since no rate variation was present in the data sets, unweighted least squares was used in the calculation of Dss.

4.4.2 An index to measure the difficulty of detecting a recombination event

As mentioned above, there are seven different recombination events in the simulation study. In order to evaluate the performance of the Dss statistic, it would be useful to be able to, in some sense, rank these events according to the relative difficulty of detecting them. A very simple index for this purpose is proposed here.

The index suggested is known as the DDR index (Difficulty of Detecting Recombination) and depends on the length of the branches connecting the sequences involved in the recombination event and the lengths of the branches from the recombination event to the tips of the tree. It is defined as

$$DDR = \text{lengths of all descendent branches affected} \div \text{lengths of all branches linking the sequences involved in the recombination event,}$$

with values lying in the range (0,1). Low values of DDR correspond to a recombination event close to the tips of the tree, which should be relatively easy to detect. Events deep in the tree have high values; it will be quite difficult for any method to find ancient recombination events due to subsequent mutations obscuring the signal.

For the ET recombination event at the first depth, there are two descendent branches, each of length 0.05 (the exterior branches leading to sequences 1 and 8). Thus, the numerator of DDR is 0.1. The denominator is the sum of the lengths of all branches linking sequences 1 and 8; this is 0.8 leading to a DDR value of 0.125. The most ancient ET recombination event (that which occurs at the fourth depth) affects all

the sequences in the data set. Therefore, the denominator of DDR is the sum of all the branch lengths (2.2). The numerator, being the amount of nucleotide substitution along all affected branch lengths, is 2.1, leading to a very high DDR value (0.955). Values of DDR for the HT recombination events range from $0.1/0.6 = 0.167$ (first depth recombination event) to $0.9/1 = 0.9$ (third depth recombination event).

The DDR index omits many factors which, more than likely, affect the ease at which any method can detect recombination. For example it would be more difficult to detect recombination in a data set with the same non-recombinant topology as Figure 4.4, but with considerably longer branch lengths. Variable branch lengths within a tree may also contribute to making the problem more difficult. Nevertheless, the simple DDR index is useful in that it does allow some quantification of the problems posed by the data sets in this simulation study

4.4.3 Evaluating the results of the simulation study

The algorithm described in this chapter for detecting recombination is a graphical method so, in practice, a plot of the Dss values for a given data set will be examined for large peaks which suggest potential recombination breakpoints. A further possible step is to consider the first differences (see **4.7.2**). However, a more automatic approach to the analysis would be preferred for the simulation study; this is both for reasons of time management and also to avoid the problem of subjectivity in the analysis. The technique used in the simulation study is described below; there may, of course, be many other possibilities.

Consider the large peaks in Dss values which are indicative of a recombination breakpoint. The points in these peaks are both large, and are surrounded by large values. Therefore, if it is possible to develop a test which finds all such points, then it should automatically find large peaks.

Firstly, a definition is needed of a large Dss value. This may be found using the two hundred data sets containing no recombination event. For each data set, the set of 201 Dss values were found. These were used to generate an empirical distribution for large Dss values. Since there can be sudden jumps in the Dss values due to random noise, a simple smoothing algorithm is applied and the largest smoothed Dss value from each data set forms part of the empirical distribution. The smoothing itself is extremely simple – each smoothed Dss value is the average of a window of 20 raw Dss values ($\sim$10% of the total number of Dss values). This yields two hundred smoothed Dss values, which may be ordered to give the empirical density function for large Dss values for this particular data set, under the hypothesis of no recombination. If an observed smoothed Dss value is greater than T points of this empirical distribution, then its p-value is $(200 - T)/200$.

To find the large points in a set of $\{Dss_i\}$, the raw values are initially smoothed as above. The p-value of each smoothed point is found using the empirical distribution. Smoothed Dss values with a p-value less than or equal to 0.01 are considered to be significantly large.

The next problem is to find the beginning and end of any peaks observed in the data, since the number of peaks carries information on the number of recombination breakpoints. The locations of the large $\{Dss_i\}$ were recorded in ascending order and the beginning and end of unbroken sequential observations noted. This gives the location of any peaks, although it is likely to be conservative, since non-significant Dss values may form the lower parts of the peaks. The location of the highest (smoothed) Dss value in each peak was also noted, since this is a possible estimate of the location of a recombination breakpoint.

The data were analysed using functions programmed in S-Plus. The output consisted of the beginning and end of each significant peak found, as well as the highest point within the peak. At that point the output was analysed manually. The presence of peaks in the correct places was noted, as was the existence of peaks in nonrecombinant regions (recorded as a *false peak*). Some peaks had significantly high smoothed Dss values entirely on one side of a peak; it was decided arbitrarily to count such peaks as being in the correct place if the nearest endpoint to the limit of the recombination event was within 50 base pairs.

4.4.4 Results of the simulation study

Tables 4.1 and 4.2 below give the results of the simulation study. The tables give the percentages of cases in which one or two peaks were found, and also display the number of data sets containing false peaks. The value of DDR is also given for each recombination event; this gives an indication of prior beliefs about the ease of detecting the various events.

From Tables 4.1 and 4.2, it appears that the Dss statistic does perform well at detecting recombination, and, as an aside, that DDR is a reasonable measure of the difficulty of detecting different recombination events, at least for this simulation study. For the first two depths of the ET recombination event (DDR values of 0.125 and 0.375), there are two peaks in the correct places. There is a low percentage of data sets with significant peaks at the wrong locations (only 4%). The Dss method also performs quite well at the third depth ($DDR = 0.625$). While this recombination event occurs quite deep in the tree, the sequences involved are relatively diverged from each other. There is very little significant at the fourth depth, but it would be very surprising if there were, as the amount of divergence in the taxa before the recombination event occurred is very small.

Table 4.1: Results for the ET type of recombination event

	1^{st} depth	2^{nd} depth	3^{rd} depth	4^{th} depth
DDR	0.125	0.375	0.625	0.875
2 peaks	100%	100%	65%	0%
1^{st} peak only	–	–	17%	2%
2^{nd} peak only	–	–	12%	3%
false peak	2%	2%	6%	–
ave highest smoothed point in 1^{st} peak	999 (945,1055)[a]	1008 (935,1085)	1034 (835,1105)	955 (915,995)
ave highest smoothed point in 2^{nd} peak	1506 (1445,1575)	1492 (1415,1605)	1470 (1365,1575)	1485 (1475,1565)
ave smoothed width of 1^{st} peak	354 (260,610)	309 (120,410)	210 (40,480)	145 (140,150)
ave smoothed width of 2^{nd} peak	347 (280,450)	304 (50,400)	226 (50,370)	143 (140,150)

[a]Figures in brackets give the range of values observed in the simulation study of the quantity above

Table 4.2: Results for the HT type of recombination event

	1^{st} depth	2^{nd} depth	3^{rd} depth
DDR	0.167	0.500	0.833
2 peaks	100%	62%	0%
1^{st} peak only	–	19%	4%
2^{nd} peak only	–	12%	0%
false peak	4%	1%	2%
ave highest smoothed point in 1^{st} peak	989 (905,1045)[a]	993 (865,1085)	1067 (1035,1115)
ave highest smoothed point in 2^{nd} peak	1511 (1465,1575)	1508 (1375,1595)	– –
ave smoothed width of 1^{st} peak	284 (160,390)	170 (10,300)	157 (80,260)
ave smoothed width of 2^{nd} peak	283 (160,380)	175 (20,340)	– –

[a]Figures in brackets give the range of values observed in the simulation study of the quantity above

The results for the HT recombination events in Table 4.2 are similar to those for the ET events, with a HT event at the first depth corresponding to an ET event at the second depth, in terms of the amount of divergence that has occurred between the sequences before the recombination event. The *Dss* algorithm successfully detects the breakpoints in a large number of data sets at the first (100%) and second (two breakpoints are found in 62 data sets, one breakpoint is found in 31 data sets) depths. The sequences involved in the recombination event at the third depth had not diverged to a great extent prior to the recombination event, so the poor performance of *Dss* here is again not surprising.

As noted above, the values of the DDR index are in approximate agreement with the simulation results – the lower the value of DDR, the easier it is, in general, to find the recombination event.

For both types of recombination event, the average width of the peaks decreases as the event moves deeper into the tree. This is unsurprising, as an event deep in the tree has had more time to accumulate mutations which obscure the signal from the recombination event.

Therefore, the conclusion from this simulation study is that the *Dss* statistic appears to have the potential to be a useful tool for biologists analysing a data set which they suspect may contain recombination. To further test this notion, the *Dss* algorithm is applied to a couple of real data sets, with known recombination events.

4.5 Examples of *Dss* applied to some real data sets

A DNA data set consisting of the *argF* gene for eight different *Neisseria* strains is now analysed using the *Dss* statistic. The strains and their accession numbers are: *N. gonorrhoeae*, X64860; *N. meningitidis*, X64861 and X64866; *N. cinerea*, X64869; *N. polysaccharea*, X64870; *N. lactamica*, X64871; *N. flavescens*, X64872 and *N. mucosa*, X64873. This data set was used by Zhou and Spratt (1992) to detect recombination in *N. meningitidis*. The data were extracted from the EMBL/GenBank/DDBJ database using their accession numbers to identify the particular sequences and aligned using the Clustal W automatic multiple alignment program (version 1.6, Thompson et al., 1994), taking the default options. The numbering scheme for the bases is that used by Zhou and Spratt (1992). Therefore, the 787 bp alignment starts at 296 bp and ends at 1083 bp.

A window of 400 bp was used to calculate the $\{Dss_i\}$, moving in increments of 2 bp each time. Following Zhou and Spratt (1992), the Jukes-Cantor model of nucleotide substitution was used. Some of the pairwise distances are small (< 0.1) so unweighted least squares is used to calculate the *Dss* statistic. The set of values calculated using weighted least squares (power=2) is also shown to further illustrate the effect of such

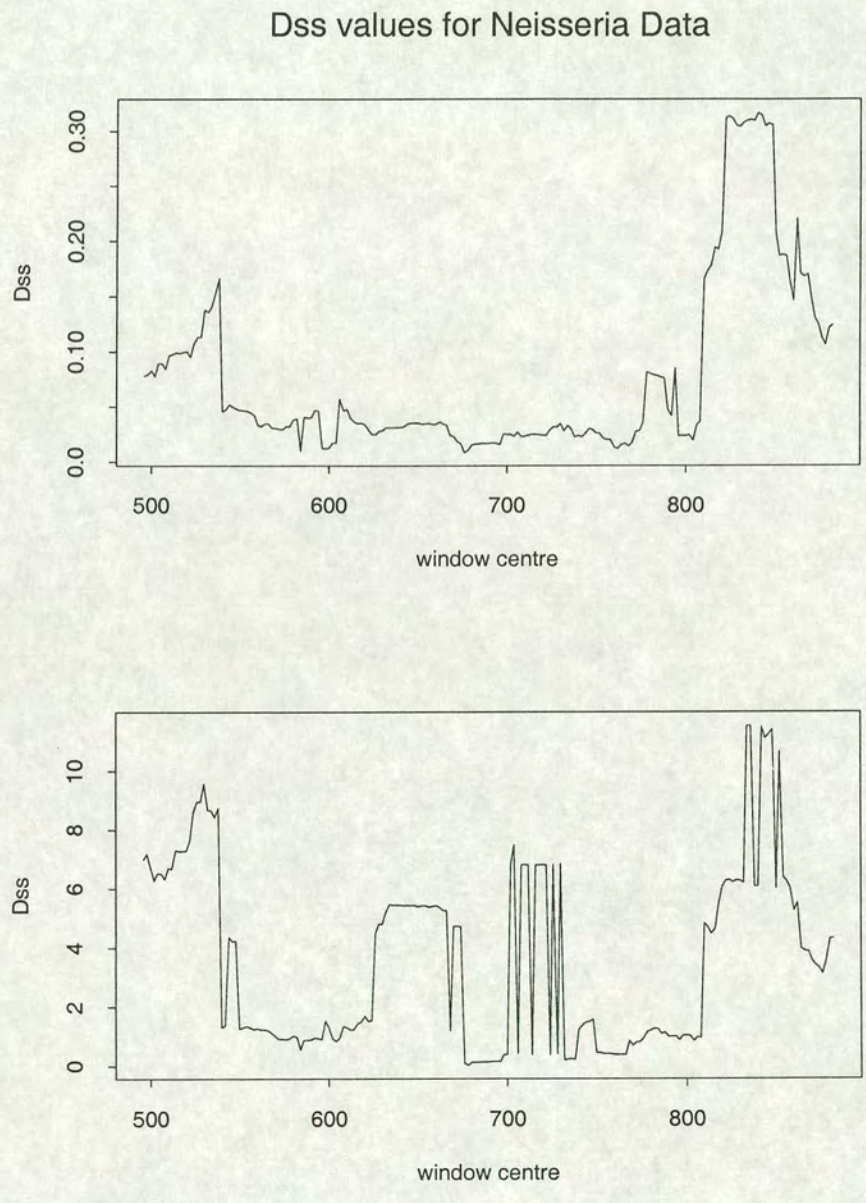


Figure 4.5: Analysis of the *Neisseria* data. *Top*: Dss values calculated using unweighted least squares, a window of 400 bp and an increment of 2 bp. *Bottom*: Dss values calculated using weighted least squares, with the same window and increment as above.

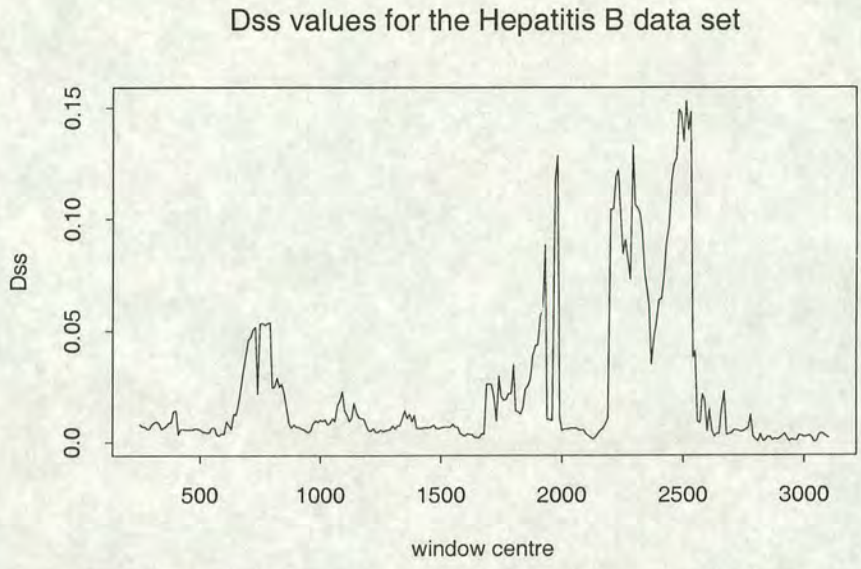


Figure 4.6: The Dss values for the Hepatitis B data set. The window size is 500 bp, with an increment of 10 bp.

short distances on Dss. The resulting plots of the Dss values are shown in Figure 4.5.

From the top graph of Figure 4.5, it appears that there are three distinct peaks, with central points located approximately at 535 bp, 787 bp and 830 bp. This suggests that the data should be split into four subsets: beginning (296 bp) to 535 bp; 536 bp to 785 bp; 786 bp to 830 bp and 831 bp to end (1083 bp). This subdivision is in good agreement with the findings of Zhou and Spratt (1992), who report anomalous regions between 296 and 497 bp, and between 803 and 833 bp. The former was found to be a recombination event; it was not known whether the latter was a recombination event, or another type of anomalous region.

The short branches effect on weighted least squares is clearly illustrated in the lower graph of Figure 4.5. A peak is present at position 625-670 bp (a region with no known recombination event), and in other places the $\{Dss_i\}$ fluctuate wildly (700-730 bp and 835-850 bp). In these regions of the data set, some of the pairwise distances are very small; indeed for some windows there is no change at all between some of the sequences. These small distances lead to the inflated values of Dss observed.

A second example consists of ten strains of the Hepatitis B virus, a subset of the data set used by Bollyky et al. (1996). The sequences used here include two recombinant strains (accession numbers D00329 and X68292), and eight nonrecombinant strains (V00866, M57663, D00330, M54923, X01587, D00630, M32138 and L27106). The Dss values were calculated using a window of 500 bp, which slides along in steps of 10 bp. Distances were again calculated using the Jukes-Cantor model of nucleotide substitution, and since the strains are quite closely related, unweighted least squares was used.

```
          Options for Dss Method                value
          ----------------------

* in tree*.phy file which contains the data        1       N
no of base pairs in each sequence                 0       L
the length of the window to be used               500     W
the size of the increment between windows         10      I
the method to be used (nj or ls)                  ls      M
the power to be used for Least squares            0.00    P
the type of data: dna or prot                     dna     D
the model of evolution (jc, k2p, ml or jn)        jc      E

Enter in a letter to choose an option to change
or enter Y if you're happy with the current options:
```

Figure 4.7: The initial text menu for the TOPAL package.

The results are shown in Figure 4.6. Four peaks are observed in the *Dss* values, occurring approximately at 730 bp, 1970 bp, 2250 bp and 2480 bp. Elsewhere the *Dss* values are small, suggesting no further recombination events. These results mirror those of Bollyky et al. (1996) who report two recombination events, one spanning positions 735 to 2370 and the other 2014 to 2203.

4.6 Software to implement the *Dss* algorithm

While the *Dss* statistic has been defined and extensively used above, the computer programs used to calculate it have not yet been described. From the definition of the statistic, it is clear that there is considerable computational work involved in obtaining a value of *Dss*. Therefore, creating a computer program to calculate *Dss* was a non-trivial exercise.

In order to facilitate these computations, the package *TOPAL* has been written and was used to carry out all the calculations in this chapter. This is a collection of unix Bourne shell scripts, C source code and the programs DNADIST, NEIGHBOR and FITCH from the PHYLIP package (the programs are included by the permission of J. Felsenstein). The package is available at http://www.bioss.sari.ac.uk/~frank/Genetics on the WWW and by anonymous ftp in the directory pub/phylogeny/topal from ftp://ftp.bioss.sari.ac.uk/.

The TOPAL package allows the application of the *Dss* algorithm not only to DNA data sets, but also to protein data sets, although computational time may be considerably greater. For example, for an amino acid data set containing 10 sequences, 2500 amino acids long, with a window size of 500 and an increment of 10 (201 windows in to-

```
                Options for Dss Method              value
                ----------------------

* in tree*.phy file which contains the data          1       N
no of base pairs in each sequence                   2500    L
the length of the window to be used                 500     W
the size of the increment between windows           10      I
the method to be used (nj or ls)                    ls      M
the power to be used for Least squares              0.00    P
the type of data: dna or prot                       prot    D
the model of protein evolution (pam or kimura)      pam     E

Enter in a letter to choose an option to change
or enter Y if you're happy with the current options:
```

Figure 4.8: The TOPAL menu after protein data has been selected.

tal), it took 1 hour using PAM distances and 18 minutes using Kimura distances (both computations were carried out on the UK HGMP-MRC computing facilities; Rysavy et al., 1992). TOPAL permits the Neighbor Joining method (see **2.7.5**) to be used, if desired, to estimate the topology in the first half of each window although the branch lengths are found using least squares. The trees are still evaluated using the sum of squares criterion. This Neighbor Joining approximation speeds up the computations considerably and makes it possible to calculate the *Dss* statistic for large data sets, without incurring an enormous computational burden.

TOPAL has a simple text menu interface. The initial menu is shown in Figure 4.7. To change an option, the letter on the far right must be typed. TOPAL requires an input file called tree*.phy where * is some number (e.g., tree1.phy, tree39.phy etc.). To select an appropriate value, type "N"; the program then prompts for a choice. The sequence length, window, increment and power may all be selected in the same way.

Other options toggle between choices. M selects whether the Neighbor-Joining approximation should be used to estimate the topology in the first half of the window, D toggles between protein and DNA data while E selects the model to be used for calculating the distances. Figure 4.8 shows the menu when protein data is selected. Note that different models of evolution are now available.

For DNA data, more options may be available depending on the model of evolution in use. For example, if the Kimura two Parameter (k2p) or maximum likelihood (ml – essentially the same as the Felsenstein 84 distance) distances are chosen, the transition-transversion ratio must also be specified. If the Jin-Nei (jn, see **2.7.3**) model is used, as in Figure 4.9, then the coefficient of variation must also be given (the coefficient of variation, CV, is used to estimate the shape parameter of the gamma distribution, a,

```
        Options for Dss Method                    value
        ----------------------

* in tree*.phy file which contains the data        1       N
no of base pairs in each sequence                 2500    L
the length of the window to be used               500     W
the size of the increment between windows         10      I
the method to be used (nj or ls)                  ls      M
the power to be used for Least squares            0.00    P
the type of data: dna or prot                     dna     D
the model of evolution (jc, k2p, ml or jn)        jn      E
the transition-transversion ratio                 2.00    T
the coefficient of variation                      1.50    C

Enter in a letter to choose an option to change
or enter Y if you're happy with the current options:
```

Figure 4.9: The TOPAL menu with the Jin-Nei model of nucleotide substitution.

since $CV = 1/\sqrt{a}$).

A full manual is included with the package and is also available on the WWW. This contains detailed instructions on how to run TOPAL. There is also further documentation and examples on the WWW at http://www.bioss.sari.ac.uk/~frank/Genetics.

4.7 Possible extensions and future work

While the *Dss* statistic, in its current form, has been found to perform quite well, there is, of course, room for development and improvement. Possible refinements to the *Dss* statistic are discussed below, and suggestions for carrying out statistical tests on the *Dss* values are also given.

4.7.1 Improving the *Dss* statistic

The *Dss* statistic is a useful way to quickly scan multiple sequence alignments for possible recombination events. The simulation study above (**4.4**), and the examples using real data sets (**4.5**) confirm this. Nevertheless, in the discussion of the properties of *Dss* (**4.3**), it was observed that *Dss* is sensitive to other factors as well as recombination. Thus, there is scope to refine *Dss* to take account of other heterogeneities in a data set so that these will not be confounded with recombination.

The *Dss* statistic is currently a simple function of the four sums of squares values obtained from the moving windows going forwards and backwards. As it stands, the magnitude of *Dss* depends on the tree length and thus, changes in the lengths of the tree along an alignment (even if no change in the branching order occurs) can have an

effect on Dss (see **4.3.2**). It is possible that an alternative, improved weighting of these four sums of squares values could be found. There may also be a way of standardising the values of Dss for tree length, other than using weighted least squares which has the problem of disproportionate effects of variation in short branch lengths. Suggestions include dividing by the total tree length, or by the sum of the entries in the distance matrix (although early investigations suggested the latter approach was not particularly useful).

Another possibility might be to modify the distance matrix in some way to lessen the effect of small distances on weighted least squares. Multiplication of distances by a constant greater than one would still preserve their relative orderings, but might eliminate the effect of short branches. A suitable approach for pairwise distances of zero would have to be found. A simple linear transformation, involving both a location and a scale change, might be sufficient. This point and those mentioned above require further study.

4.7.2 Statistical tests for significant Dss values

This method is essentially being proposed as a graphical method to detect recombination; it does not claim to give any definitive answers about the presence of recombination in a data set. Nevertheless, it would be useful to have an approximate statistical test which could, to some extent, measure the degree of confidence in the results. Two possibilities are considered below: a simple test based on first differences (using elementary time series principles); the second possibility is parametric bootstrapping.

Since the Dss values are obtained from windows which overlap, they are highly positively correlated. If the increment size is small relative to the window size then Dss_i will be approximately independent of Dss_{i-2} given Dss_{i-1}. This suggests that the first differences of the $\{Dss_i\}$ may be used to test the significance of high Dss values.

In the absence of recombination, the first differences,

$$\Delta_i = Dss_{i+1} - Dss_i$$

should be approximately independent. Making the further assumption that the $\{\Delta_i\}$ are approximately normally distributed, confidence intervals (e.g., 95% and 99%) for the $\{\Delta_i\}$ may be constructed. If the first differences show a greater spread in a region, and there are some significantly large points, then this suggests that the corresponding peak does mark a recombination breakpoint in the data.

The Dss values from the simulated data set used in Figure 4.1 are shown again in Figure 4.10, together with the first differences. A greater spread in the first differences is noted in the areas of the alignment where the the recombination breakpoints occur;

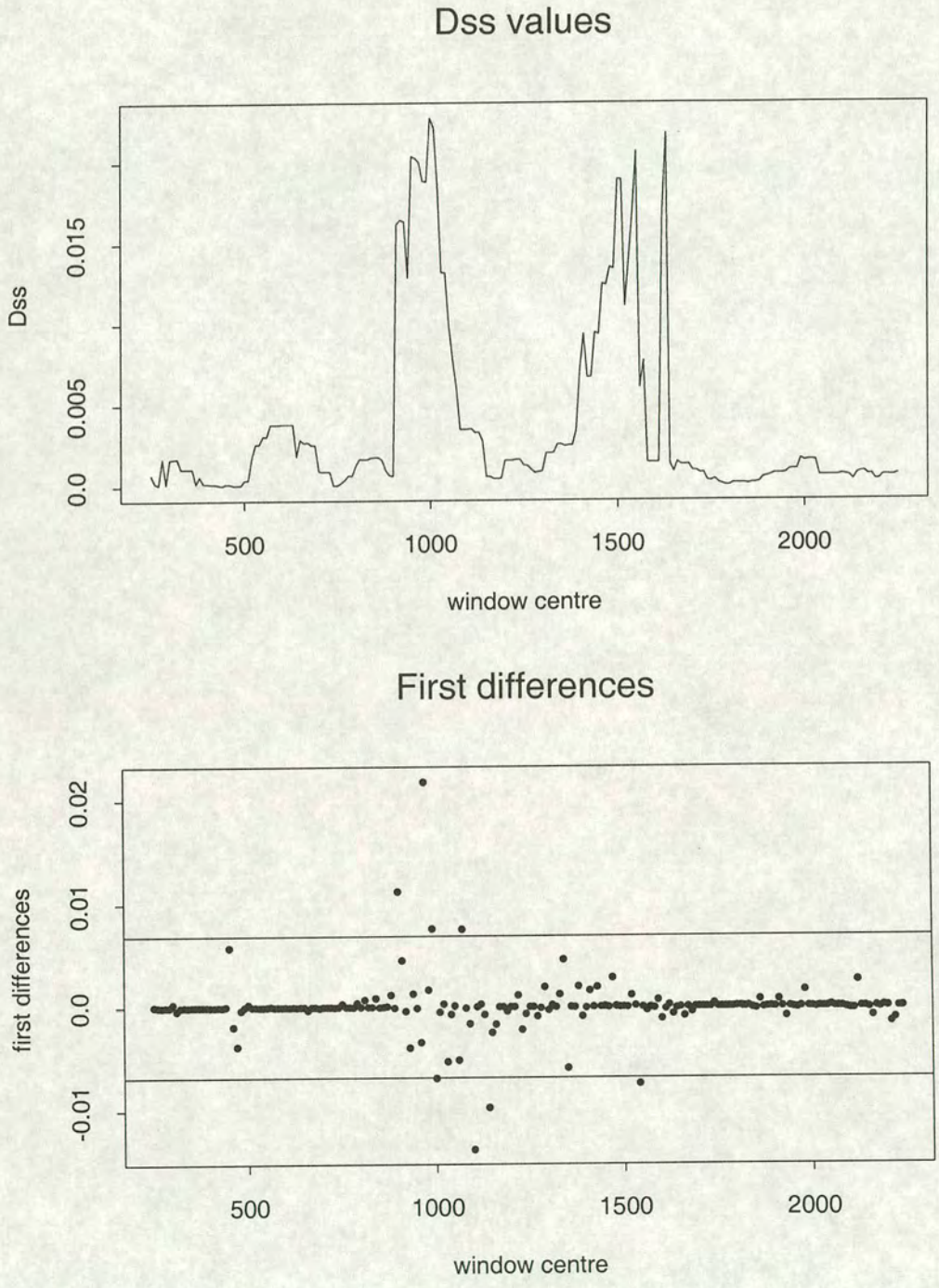


Figure 4.10: The *Dss* values from the simulated data set in Figure 4.1 are shown again, along with the corresponding first difference (bottom graph). The horizontal lines in the bottom graph mark the approximate 99% confidence interval.

some of these first differences are significant at the 99% level. There are also some large first differences caused by random noise in the data set, rather than a recombination event; these do not appear to be significant.

This test does seem to perform quite well. While peaks and troughs are observed in *Dss* values from data sets with no recombination, the increase or decrease from one *Dss* value to the next tends to be quite small, so the first differences are generally of low magnitude. However, in a data set with recombination, large jumps in the *Dss* values are often observed, leading to large first differences which are picked up by this test.

A more rigorous test would be to apply some form of parametric bootstrapping to the data set similar to that used to evaluate the results of the simulation study in **4.4**. This would involve simulating many data sets under the hypothesis of no recombination and finding a distribution for large *Dss* values. The problem with this approach is the choice of tree to be used to generate the data sets. The tree estimated on the observed data set, if it contains a recombination event, will be an average of the different local trees. Therefore, the simulation will not be carried out under the true null hypothesis. It has also been noted above that many factors (tree length, branch lengths etc.) affect the *Dss* statistic, so simulating from a slightly incorrect tree is likely to produce values of *Dss* on an incorrect scale, resulting in too conservative or too liberal a test. However, if it were possible to standardise *Dss* over different tree shapes and lengths as discussed above, such an approach could be both feasible and useful.

The *Dss* statistic and the accompanying computer package, TOPAL, have the potential to become useful tools for biologists. It is hoped that future work will yield improvements, which should increase their usefulness.

Chapter 5

A Bayesian Approach to Modelling Recombination

In the previous chapter, a graphical approach was developed for detecting recombination in DNA alignments. This method, using the *Dss* statistic, is useful as an initial tool in a statistical analysis of a DNA data set as it can quickly scan an alignment for evidence of recombination. However, this algorithm merely detects the possible presence of a recombination event; it makes no attempt to model it.

A Bayesian analysis of topology change due to recombination along a DNA sequence alignment is presented in this chapter. For computational reasons, data sets are restricted to four sequences. The chapter opens by examining the motivation behind this work: the likelihoods for each possible topology at each site. The theory of Hidden Markov models is described since this plays a vital role in the methodology. A Bayesian analysis of recombination is then presented. The performance of this method and the sensitivity of the results to the choice of prior is assessed for simulated data sets. An example using some of the *Neisseria* sequences described in **4.5** is given. Finally, some possible extensions are discussed. The relationship between this Bayesian approach and the parsimony-based method described by Hein (1993, see **3.2**) is also discussed since this suggests a direction for future work.

5.1 Motivation

Consider a set of four DNA sequences, one of which has incorporated genetic material from another at some point in the past. Thus, this recombination event will result in a change of topology in the affected region. For a set of four sequences, there are only three possible unrooted topologies; therefore it seems reasonable to calculate the likelihood at each site for each possible topology and compare these. Labelling the three possible topologies as 12 (i.e., sequences 1 and 2 cluster together), 13 (sequences 1 and 3 together) and 14 (sequences 1 and 4 together), then there should be regions in which one topology corresponds to the highest likelihood at each site. If one topology is

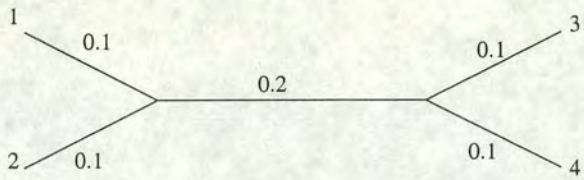


Figure 5.1: Tree used for simulating the data.

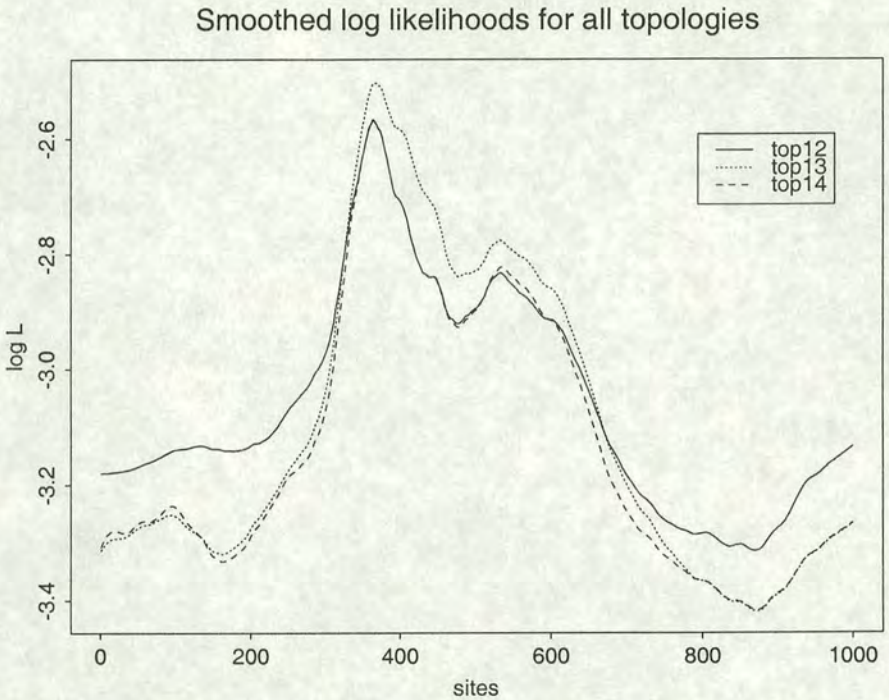


Figure 5.2: Smoothed log likelihoods for each of the three possible topologies for the simulated data set, described in the text. Top1j represents the topology with sequence 1 clustering with sequence j, $j = 2, 3, 4$, the other two sequences forming another group.

dominant for a reasonably long stretch of sites followed by another one being dominant in an adjacent region, this would suggest that a recombination event has occurred, resulting in a change of topology.

To test this idea, a data set of four sequences, 1000 nucleotides long, was simulated according to the tree in Figure 5.1. The Kimura two Parameter model of evolution was used, with a transition-transversion ratio of 2 (see **2.5**). The data were simulated along the interior branch, and then along the four outer branches until 90% of the nucleotide substitutions had occurred. At that point, a central subsequence (301–700 nucleotides) of sequence 3 replaced the corresponding subsequence in sequence 1. The nucleotide substitution process then resumed for the remaining 10% of the branch length.

Once the data set was obtained, the likelihoods at each site for each topology were calculated. Due to the possible problems of conflicting information on branch lengths when using the entire data set (see **5.3.2** for more details), the data set was broken down into subalignments of twenty sites, and the best tree for each topology, and thus, the

Table 5.1: The frequencies of the largest likelihood corresponding to topologies 2, 3 and 4 in each of the three regions of the simulated data set

	1–300	301–700	701–1000
top. 2	209	84	204
top. 3	30	254	24
top. 4	61	62	72

log likelihoods for each site of the subsequence were found using the PHYLIP program, DNAML (modified to output the log likelihoods at each site). These were then plotted against the corresponding sites in Figure 5.2. For ease of interpretation, a smoothing algorithm (the *supsmu* algorithm in S-plus) was used to smooth the data.

Due to the recombination event in the simulated data set, it would be expected that the topology placing sequences 1 and 2 together should have the highest site likelihoods for sites 1–300 and 701–1000. In between (301–700 bp), the topology with sequences 1 and 3 clustering together should have the highest likelihoods, or equivalently log likelihoods. This is, indeed, the case from Figure 5.2. The cross-over points are not located exactly at 301 and 700 nucleotides; this may be due to statistical noise, or the smoothing algorithm or a combination of both.

Another way to examine this data is to look at the topology corresponding to the largest likelihood at each site. This may be easily found using the S-plus statistical package. The topology with sequences 1 and 2 together is represented by 2; that with sequences 1 and 3 together is depicted by 3 while 4 stands for the topology with sequences 1 and 4 as neighbours. The output in Figure 5.3 is then obtained.

The data set has been split into three regions, corresponding to the exact recombination breakpoints. By simply looking at each part of the data set, it is seen, as expected, that topology 2 occurs most frequently in the first and last regions, while topology 3 is the most common in the second region. Table 5.1 gives the frequencies of each of the three topologies in each of these regions of the sequence. From the simulation design, it is known that the first 300 sites all have the same topology (top. 2) so other topologies in this region corresponding the largest likelihood are simply the result of statistical noise. These topologies tend to occur in short runs. Since it is known that these are not due to recombination, they give an idea of the level of noise which may be present in data sets. Similar conclusions may be drawn for the other two subsets of the data (301–700 nucleotides and 701–1000 nucleotides).

The statements above are based on two components. Firstly, the likelihoods give information on the most likely topology at each site. Secondly, existing knowledge about recombination events is used to decide when a true change in topology is most likely to have occurred as opposed to noise. This is, in essence, a Bayesian approach,

Sites 1–300

```
  [1] 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 4 4
 [31] 2 4 2 2 2 2 2 2 4 2 2 2 2 2 3 2 2 2 2 2 2 2 3 2 3 2 2 2 2 3
 [61] 2 2 4 2 2 4 3 2 2 2 2 2 2 2 2 2 2 2 2 4 4 2 2 4 2 2 2 2 2 2
 [91] 3 2 2 2 2 2 2 2 4 3 2 3 2 4 2 2 4 2 2 2 2 2 2 4 4 2 2 2 2 2
[121] 2 4 4 2 2 4 4 4 4 2 4 4 2 4 4 4 4 2 4 4 2 2 3 2 2 2 2 2 3 2
[151] 2 4 2 2 2 3 2 4 2 2 4 4 4 4 4 4 4 2 4 4 4 4 4 3 4 4 4 2 4 4
[181] 3 2 2 2 2 2 2 2 3 2 2 2 2 2 2 3 3 2 2 2 3 2 2 2 2 2 4 2 2 2
[211] 2 4 2 4 2 2 2 2 4 2 2 2 4 2 3 4 2 2 2 2 2 4 2 2 3 2 3 2 2 2
[241] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 3 2 2 2 2 3 2 2 2 3 2 2 3 2
[271] 2 4 3 2 3 4 2 2 2 2 2 4 2 2 2 4 2 2 2 2 2 2 4 2 4 3 2 2 2 4
```

Sites 301–700

```
[301] 3 3 3 2 3 4 3 3 3 3 3 4 3 3 3 3 3 3 2 4 3 3 3 3 4 4 3 3 3 3
[331] 3 3 3 3 2 3 3 3 4 4 2 3 3 3 3 3 3 3 3 3 3 2 3 3 2 3 3 3 3 3
[361] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 2 3 2 3 3 3 3
[391] 3 3 3 3 4 3 3 4 3 3 2 3 3 3 3 3 2 3 3 3 3 3 3 3 2 3 3 3 2 3
[421] 3 3 2 3 2 2 3 3 3 3 4 3 3 3 3 2 3 3 3 3 3 4 3 3 3 3 3 3 3 3
[451] 4 3 2 2 3 3 3 2 3 4 3 4 4 2 3 3 3 3 3 4 4 3 4 4 3 4 3 2 4 3
[481] 3 3 3 3 3 3 3 4 4 3 4 3 3 3 4 3 4 3 3 3 2 2 2 3 2 2 3 3 3 3
[511] 2 3 3 3 3 3 3 3 3 3 3 3 2 3 2 3 2 2 3 3 2 3 3 2 3 3 2 2 3 3
[541] 2 2 3 2 4 2 2 2 3 3 3 2 2 2 2 2 3 3 3 2 4 3 4 4 4 4 3 4 2 3
[571] 3 4 4 4 4 4 4 4 4 2 4 3 3 3 3 3 3 2 3 3 3 2 3 3 3 2 3 4 4 4 3
[601] 3 3 3 3 2 3 3 3 3 3 4 4 3 3 3 3 3 3 3 3 3 2 3 2 2 3 2 3 3 2 3
[631] 3 3 3 3 2 3 2 3 3 3 4 3 3 3 3 3 3 3 3 3 3 3 3 4 3 3 2 3 3 3
[661] 2 4 3 3 3 3 4 3 4 4 4 4 2 3 3 4 3 4 3 3 4 3 4 4 4 3 3 3 3 3
[691] 3 4 3 2 3 4 3 3 3 3
```

Sites 701–1000

```
[701] 2 2 2 2 4 2 2 2 2 2 4 2 2 4 2 2 4 2 3 2 2 2 2 4 2 2 2 2 4 2
[731] 4 2 2 2 2 2 2 2 2 2 4 3 2 2 2 4 2 2 2 2 4 3 4 4 3 2 2 2 2 2
[761] 2 2 4 2 2 2 2 2 2 2 2 2 2 3 4 4 2 2 4 2 2 4 4 3 4 4 4 4 2 4
[791] 4 4 4 4 4 2 3 2 4 4 2 2 2 2 3 2 4 2 2 3 4 4 2 2 2 2 2 2 2 2
[821] 4 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 4 2 2 2 4 2 2 4 4 2 2
[851] 2 4 2 2 2 2 2 2 4 2 3 2 4 2 2 2 2 2 2 2 4 2 2 2 4 4 2 2 2 2
[881] 3 2 2 2 2 4 3 2 2 2 2 2 2 3 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 3
[911] 2 2 2 2 2 2 2 3 2 2 2 4 4 3 4 4 4 4 4 4 2 3 4 4 4 4 3 4 4 4 3
[941] 4 2 2 3 2 2 2 2 4 2 2 2 2 2 2 3 4 2 4 2 2 4 2 2 2 2 2 4 2 2
[971] 2 2 2 2 2 2 2 2 2 2 4 4 2 2 2 2 2 2 2 3 2 4 2 2 4 2 4 2 4 2 2
```

Figure 5.3: Topologies corresponding to the largest likelihood at each site. The numbers in square brackets denote the position in the alignment. Each position has an associated integer value (2, 3 or 4) corresponding to the topology with the highest likelihood at that site.

combining likelihood and prior information, and suggests how a Bayesian approach to modelling topology changes due to recombination may be developed; this is described later in **5.3**. The computations that must be carried out rely on the theory of Hidden Markov models. Hence, a brief overview of this subject area is given in the next section.

5.2 Theory of Hidden Markov models

Hidden Markov models are found in a number of fields of science. For example, the problem of signal processing may be formulated as follows:

$$Y_i = X_i + \epsilon_i \tag{5.1}$$

where Y_i is the observed signal;

X_i is the actual signal broadcast, assumed to be a Markov process;

ϵ_i is a noise process.

Here the $\{X_i\}$ constitute a Hidden Markov model since they cannot be observed directly, but can only be inferred from the observed signal, the $\{Y_i\}$.

Another example of a Hidden Markov model in the time series field has been described by MacDonald and Zucchini (1997, p. 55). Suppose the number of occurrences of a particular event in a fixed period of time is being counted. This is often modelled as a Poisson process with mean λ and variance λ. However, such count data is often over-dispersed (the variance exceeds the mean). There may also be serial dependence (for example, the number of epileptic seizures in one patient on successive days). An alternative approach to a Poisson process is to suppose that each observation is generated by one of two Poisson distributions with means λ_1 and λ_2 respectively where the choice of distribution (i.e., the value of the mean) is made by another random mechanism - the parameter process. Letting $P(\lambda_i)$ represent the Poisson process with parameter λ_i, then the parameter process selects $P(\lambda_1)$ with probability δ_1 and $P(\lambda_2)$ with probability $\delta_2 = 1 - \delta_1$. This model demands that the variance exceeds the mean since the variance is given by $\delta_1\lambda_1 + \delta_2\lambda_2 + (\lambda_1 + \lambda_2)^2\delta_1\delta_2$.

This model consists of two layers - the outcome (i.e., the counts observed) and the parameter process which cannot be observed, merely inferred from the outcome. If the parameter process is assumed to be a Markov chain, then the resulting process of counts allows for serial dependence and is an example of a Hidden Markov model.

A substantial amount of study has been devoted to Hidden Markov Models. Below the theory behind these models is described. Details on how to carry out certain computations efficiently are also given; in particular the Viterbi algorithm is described. This is a dynamic programming method which finds the most likely sequence of states in the Hidden Markov model.

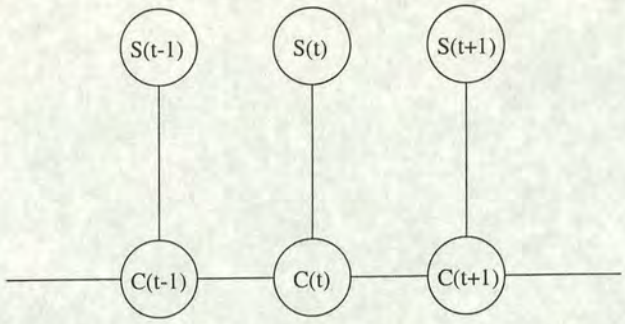


Figure 5.4: Conditional independence graph of a Hidden Markov model.

5.2.1 The model

Hidden Markov models have been frequently used and described in the speech processing literature (for example, see Juang and Rabiner, 1991). They have also been described by MacDonald and Zucchini (1997). Following the latter reference on p. 66, let $\{C_t : t \in \mathbb{N}\}$ be an irreducible, homogeneous discrete-time stationary first-order Markov chain on the state space $\{1, 2, \ldots, m\}$ with transition probability matrix $\mathbf{P}$, containing elements p_{ij} where

$$p_{ij} = \text{Prob}(C_t = j | C_{t-1} = i). \tag{5.2}$$

Since $\{C_t\}$ is stationary and irreducible, there exists a unique, strictly positive stationary distribution denoted by $\mathbf{f} = (f_1, f_2, \ldots, f_m)$.

Now consider another random process $\{\mathbf{S}_t : t \in \mathbb{N}\}$. Conditional on $C^{(N)} = \{C_t : t = 1, 2, \ldots, N\}$, N being the total number of observations, the random variables $\{\mathbf{S}_t : t = 1, 2, \ldots, N\}$ are mutually independent. Also suppose that

$$\text{Prob}(\mathbf{S}_t = \mathbf{s} | C_t = i) := {}_t\pi_{si} \tag{5.3}$$

are the state-dependent probabilities. If these do not depend on the time t, then the subscript t may be omitted. Since this will be the case for the application described in this chapter, the subscript t in (5.3) will be left out from now on.

This model may be represented by a conditional independence graph, like that shown in Figure 5.4. From it, the independence of the $\{\mathbf{S}_t\}$ given the $\{C_t\}$ may be easily seen. The graph also shows the conditional independence of C_{t+1} and C_{t-1} given C_t, which is, of course, the Markov property.

Given a Hidden Markov model as described above, calculating the likelihood (proportional to the sum of the probabilities of all possible configurations of the state process, $\{C_t\}$), seems an intractable calculation, as does finding the configuration of states $\{C_1, \ldots, C_N\}$ which contributes the most to the likelihood. Fortunately, this is not the case. Various algorithms exist which allow these calculations to be carried out efficiently. For example, the forward and backward probabilities may be used to find

the likelihood, while the maximum likelihood estimate can be found using the Viterbi algorithm.

To develop these algorithms, some properties of Hidden Markov models are required. These are stated in MacDonald and Zucchini (1997, p. 59) and proved in their Appendix A (pp. 203–206). These properties, together with their proofs, are given below.

5.2.2 Properties of Hidden Markov models

The following four properties are used to facilitate computations for Hidden Markov models. Note that, for ease of notation, the event $\mathbf{S}_t = \mathbf{s}_t$ is denoted by $\mathbf{S}_t$.

Property 1 For $t = 1, 2, \ldots, N$

$$\text{Prob}(\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_N | C_t) = \text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_t | C_t) \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C_t).$$

If $t = N$, then the convention that

$$\text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C_t) = 1$$

is used.

Property 2 For $t = 1, 2, \ldots, N-1$

$$\text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_N | C_t, C_{t+1}) = \text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_t | C_t) \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C_{t+1}).$$

Property 3 For $1 \leq t \leq l \leq N$

$$\text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_t, \ldots, C_l) = \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_l).$$

Property 4 For $t = 1, 2, \ldots, N$

$$\text{Prob}(\mathbf{S}_t, \ldots, \mathbf{S}_N | C_t) = \text{Prob}(\mathbf{S}_t | C_t) \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C_t).$$

In general, the steps used to prove these properties are:

(a) express the probability of interest in terms of probabilities conditional on $C^{(N)} = (C_1, \ldots, C_N)$, i.e., conditional on all of $C_1, \ldots, C_N$;

(b) use the fact that, conditional on $C^{(N)}$, the random variables $\mathbf{S}_1, \ldots, \mathbf{S}_N$ are independent, with the distribution of each $\mathbf{S}_t$ depending only on the corresponding C_t;

(c) use the Markov property of $\{C_t\}$ if necessary.

To establish these relationships, Property 1 is firstly proved using the three propositions below. Then the fourth property is derived from it. The second and third properties follow on from this.

Proposition 1: For all integers t and l such that $1 \leq t \leq l \leq N$,

$$\boxed{\text{Prob}(\mathbf{S}_l, \mathbf{S}_{l+1}, \ldots, \mathbf{S}_N | C_t, \ldots, C_N) = \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_l, \ldots, C_N)}$$

Proof

The left-hand side of the above may be written as

$$\frac{1}{\text{Prob}(C_t, \ldots, C_N)} \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N, C_t, \ldots, C_N)$$

which is equivalent to

$$\frac{1}{\text{Prob}(C_t, \ldots, C_N)} \sum_{c_1, \ldots, c_{t-1}} \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C^{(N)}) \text{Prob}(C^{(N)})$$

where $C^{(N)} = C_1, \ldots, C_N$, with no summation if $t = 1$. Using (b) it is seen that

$$\text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C^{(N)}) = \text{Prob}(\mathbf{S}_l | C_l) \ldots \text{Prob}(\mathbf{S}_N | C_N)$$

which can be taken outside the summation. Since the sum reduces to $\text{Prob}(C_t, \ldots, C_N)$, the left-hand side is simply

$$\text{Prob}(\mathbf{S}_l | C_l) \ldots \text{Prob}(\mathbf{S}_N | C_N)$$

which is independent of t. The right-hand side, representing the case $t = l$ of the left-hand side equals the same expression. □

Proposition 2: For $t = 1, 2, \ldots, N-1$

$$\boxed{\text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C^{(t)}) = \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C_t)}$$

Proof

The left-hand side may be written as

$$\frac{1}{\text{Prob}(C^{(t)})} \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N, C_1, \ldots, C_t)$$
$$= \frac{1}{\text{Prob}(C^{(t)})} \sum_{c_{t+1}, \ldots, c_N} \text{Prob}(C^{(N)}) \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C^{(N)}).$$

Now

$$\frac{\text{Prob}(C_1, \ldots, C_N)}{\text{Prob}(C_1, \ldots, C_t)} = \text{Prob}(C_{t+1}, \ldots, C_N | C_1, \ldots, C_t)$$
$$= \text{Prob}(C_{t+1}, \ldots, C_N | C_t)$$

by the Markov property of the $\{C_t\}$. Also

$$\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_1,\ldots,C_N) = \text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_t,C_{t+1},\ldots,C_N)$$

by Proposition 1. The left-hand side now becomes

$$\sum_{c_{t+1},\cdots,c_N} \text{Prob}(C_{t+1},\ldots,C_N|C_t)\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_t,\ldots,C_N)$$

$$= \sum_{c_{t+1},\cdots,c_N} \text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_t,\ldots,C_N)\frac{\text{Prob}(C_t,C_{t+1},\ldots,C_N)}{\text{Prob}(C_t)}.$$

The summand may be expressed as $\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N,C_t,\ldots,C_N)/\text{Prob}(C_t)$ and the sum is thus equal to $\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N,C_t)/\text{Prob}(C_t)$ as required. □

Proposition 3: For $t = 1,\ldots,N$

$$\boxed{\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C^{(N)}) = \text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C^{(t)})}$$

Proof

Apply (b) in respect of the conditioning on $C^{(N)}$ to see that the left-hand side equals $\text{Prob}(\mathbf{S}_1|C_1)\ldots\text{Prob}(\mathbf{S}_t|C_t)$. Then apply (b) in respect of the conditioning on $C^{(t)}$ to see that the right-hand side equals the same expression.□

It is now possible to prove Property 1, that

$$\boxed{\text{Prob}(\mathbf{S}_1,\mathbf{S}_2,\ldots,\mathbf{S}_N|C_t) = \text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C_t)\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_t)}$$

for $t = 1,\ldots,N$.

Proof

Making use of the mutual independence of $\mathbf{S}_1,\ldots,\mathbf{S}_N$ given $C^{(N)}$, write the left-hand side as

$$\frac{1}{\text{Prob}(C_t)}\sum_{c_1,\cdots,c_{t-1}}\sum_{c_{t+1},\cdots,c_N}\text{Prob}(C^{(N)})\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C^{(N)})$$
$$\times\,\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C^{(N)}).$$

Using Proposition 3 this becomes

$$\frac{1}{\text{Prob}(C_t)}\sum_{c_1,\cdots,c_{t-1}}\sum_{c_{t+1},\cdots,c_N}\text{Prob}(C^{(N)})\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C^{(t)})$$
$$\times\,\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C^{(N)}).$$

Summing over $c_{t+1},\ldots,c_N$ and using Proposition 2 yields

$$\frac{1}{\text{Prob}(C_t)}\sum_{c_1,\cdots,c_{t-1}}\text{Prob}(C^{(t)})\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C^{(t)})\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_t)$$

$$= \frac{1}{\text{Prob}(C_t)}\left[\sum_{c_1,\cdots,c_{t-1}}\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t,C_1,\ldots,C_t)\right]\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_t).$$

This is equal to

$$\frac{1}{\text{Prob}(C_t)}\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t,C_t)\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_t)$$

i.e., the right-hand side.□

Property 4 states that, for $t = 1, 2, \ldots, N$,

$$\boxed{\text{Prob}(\mathbf{S}_t,\ldots,\mathbf{S}_N|C_t) = \text{Prob}(\mathbf{S}_t|C_t)\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_t)}$$

Proof

Simply sum the result of Property 1 with respect to $\mathbf{s}_1,\ldots,\mathbf{s}_{t-1}$.□

Recall that Property 2 states that

$$\boxed{\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_N|C_t,C_{t+1}) = \text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C_t)\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_{t+1})}$$

for $t = 1,\ldots,N-1$.

Proof

Following previous proofs, write the left-hand side as

$$\frac{1}{\text{Prob}(C_t,C_{t+1})}\sum_{c_1,\ldots,c_{t-1}}\sum_{c_{t+2},\ldots,c_N}\text{Prob}(C^{(N)})\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C^{(N)})$$
$$\times\,\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C^{(N)}).$$

By Propositions 3 and 1, the last two factors reduce to $\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C^{(t)})$ and $\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_{t+1},\ldots,C_N)$ respectively, yielding

$$\frac{1}{\text{Prob}(C_t,C_{t+1})}\sum_{c_1,\ldots,c_{t-1}}\sum_{c_{t+2},\ldots,c_N}\text{Prob}(C^{(N)})\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C^{(t)})$$
$$\times\,\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_{t+1},\ldots,C_N).$$

The Markov property of C_t is then used, and followed by some routine manipulations of conditional probabilities, it emerges that the left-hand side is equal to

$$\begin{aligned}
&\sum_{c_1,\ldots,c_{t-1}}\sum_{c_{t+2},\ldots,c_N}\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C^{(t)})\\
&\quad\times\text{Prob}(C_1,\ldots,C_{t-1},C_{t+2},\ldots,C_N|C_t,C_{t+1})\\
&\quad\times\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_{t+1},\ldots,C_N)\\
&=\sum_{c_{t+2},\ldots,c_N}\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C_t)\text{Prob}(C_{t+2},\ldots,C_N|C_t,C_{t+1})\\
&\quad\times\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_{t+1},\ldots,C_N)\\
&=\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C_t)\sum_{c_{t+2},\ldots,c_N}\frac{1}{\text{Prob}(C_{t+1})}\text{Prob}(C_{t+1},\ldots,C_N)\\
&\quad\times\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N|C_{t+1},\ldots,C_N)\\
&=\text{Prob}(\mathbf{S}_1,\ldots,\mathbf{S}_t|C_t)\frac{1}{\text{Prob}(C_{t+1})}\sum_{c_{t+2},\ldots,c_N}\text{Prob}(\mathbf{S}_{t+1},\ldots,\mathbf{S}_N,C_{t+1},\ldots,C_N).
\end{aligned}$$

Upon summation this expression becomes

$$\text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_t | C_t) \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N, C_{t+1}) / \text{Prob}(C_{t+1})$$

which is equivalent to the right-hand side. □

Finally, Property 3 states that for all $t, l \in \mathbb{N}$ such that $1 \leq t \leq l \leq N$,

$$\boxed{\text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_t, \ldots, C_l) = \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_l)}$$

Proof

The left-hand side may be written as

$$\frac{1}{\text{Prob}(C_t, \ldots, C_l)} \sum_{c_{l+1}, \cdots, c_N} \sum_{c_1, \cdots, c_{t-1}} \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C^{(N)}) \text{Prob}(C^{(N)}).$$

By Proposition 1,

$$\text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C^{(N)}) = \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_l, \ldots, C_N)$$

and the above expression for the left-hand side becomes

$$\begin{aligned}
&\frac{1}{\text{Prob}(C_t, \ldots, C_l)} \sum_{c_{l+1}, \cdots, c_N} \sum_{c_1, \cdots, c_{t-1}} \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_l, \ldots, C_N) \text{Prob}(C^{(N)}) \\
&= \sum_{c_{l+1}, \cdots, c_N} \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_l, \ldots, C_N) \\
&\quad \times \left[\sum_{c_1, \cdots, c_{t-1}} \text{Prob}(C_1, \ldots, C_{t-1}, C_{l+1}, \ldots, C_N | C_t, \ldots, C_l) \right] \\
&= \sum_{c_{l+1}, \cdots, c_N} \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_l, \ldots, C_N) \text{Prob}(C_{l+1}, \ldots, C_N | C_t, \ldots, C_l) \\
&= \sum_{c_{l+1}, \cdots, c_N} \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_l, \ldots, C_N) \text{Prob}(C_{l+1}, \ldots, C_N | C_l)
\end{aligned}$$

by the Markov property of the $\{C_t\}$. Upon further manipulation of conditional properties, the left-hand side becomes

$$\begin{aligned}
&\sum_{c_{l+1}, \cdots, c_N} \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N | C_l, \ldots, C_N) \text{Prob}(C_l, \ldots, C_N) / \text{Prob}(C_l) \\
&\qquad = \frac{1}{\text{Prob}(C_l)} \sum_{c_{l+1}, \cdots, c_N} \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N, C_l, \ldots, C_N) \\
&\qquad = \text{Prob}(\mathbf{S}_l, \ldots, \mathbf{S}_N, C_l) / \text{Prob}(C_l)
\end{aligned}$$

which is equivalent to the right-hand side. □

5.2.3 Efficient calculations for Hidden Markov models

Two algorithms are now described, which make certain computations with Hidden Markov models quick and efficient. The first is the *forward-backward* algorithm and may be used to calculate the likelihood for a given model (sums the probabilities of observing the $\{\mathbf{S}_t\}$ given all the possible configurations of the process $\{C_t\}$; MacDonald and Zucchini, 1997, p. 59).

Essentially the forward-backward algorithm requires the computation of the forward probabilities, $\alpha_t(i)$, and the backward probabilities, $\beta_t(i)$, so called because they require a forward and a backward pass through the data respectively. They are defined as

$$\alpha_t(i) = \text{Prob}(\mathbf{S}_1 = \mathbf{s}_1, \ldots, \mathbf{S}_t = \mathbf{s}_t, C_t = i) \tag{5.4}$$

and

$$\beta_t(i) = \text{Prob}(\mathbf{S}_{t+1} = \mathbf{s}_{t+1}, \ldots, \mathbf{S}_N = \mathbf{s}_N | C_t = i). \tag{5.5}$$

Note that the convention that $\text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C_t) = 1$ when $t = N$ implies that $\beta_N(i) = 1$ for all i.

From (5.4), (5.5) and Property 1, it is seen that, for $t = 1, 2, \ldots, N$,

$$\begin{aligned} \alpha_t(i)\beta_t(i) &= \text{Prob}(C_t = i)\text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_t | C_t = i)\text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C_t = i) \\ &= \text{Prob}(C_t = i)\text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_N | C_t = i) \\ &= \text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_N, C_t = i) \end{aligned}$$

and so

$$\begin{aligned} \sum_{i=1}^{m} \alpha_t(i)\beta_t(i) &= \text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_N) \\ &= L_N \end{aligned} \tag{5.6}$$

where $L_N = \text{Prob}(\mathbf{S}_1 = \mathbf{s}_1, \ldots, \mathbf{S}_N = \mathbf{s}_N)$, which is, of course, the likelihood.

It is observed from (5.6) that, if it is possible to evaluate the forward and backward probabilities for all t, then there are N different ways of calculating the likelihood. For example, setting $t = N$ yields the formula

$$L_N = \sum_{i=1}^{m} \alpha_N(i)$$

which is the formula usually quoted in the speech processing literature (see Juang and Rabiner, 1991).

In order to find all the forward and backward probabilities it is firstly noted that

$$\beta_N(i) = 1 \tag{5.7}$$

and

$$\begin{aligned}\alpha_1(i) &= \text{Prob}(C_1 = i)\text{Prob}(\mathbf{S}_1 = \mathbf{s}_1 | C_1 = i) \\ &= f_i \pi_{\mathbf{s}_1 i}. \end{aligned} \tag{5.8}$$

These values are used in the recursions developed below.

Using Property 2

$$\begin{aligned}\alpha_{t+1}(j) &= \text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_{t+1}, C_{t+1} = j) \\ &= \sum_{i=1}^{m} \text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_{t+1}, C_t = i, C_{t+1} = j) \\ &= \sum_{i=1}^{m} \text{Prob}(C_t = i, C_{t+1} = j)\text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_{t+1} | C_t = i, C_{t+1} = j) \\ &= \sum_{i=1}^{m} \text{Prob}(C_{t+1} = j | C_t = i)\text{Prob}(C_t = i) \\ &\quad \times \text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_t | C_t = i)\text{Prob}(\mathbf{S}_{t+1} | C_{t+1} = j) \\ &= \sum_{i=1}^{m} \text{Prob}(\mathbf{S}_1, \ldots, \mathbf{S}_t, C_t = i) p_{ij} \pi_{\mathbf{s}_{t+1} j} \\ &= \left(\sum_{i=1}^{m} \alpha_t(i) p_{ij} \right) \pi_{\mathbf{s}_{t+1} j}, \end{aligned} \tag{5.9}$$

this recursion being valid for $1 \leq t \leq N - 1$.

To set up a recursion for the backward probabilities, use Property 3 with $l = t + 1$ and Property 4:

$$\begin{aligned}\beta_t(i) &= \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C_t = i) \\ &= \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N, C_t = i)/\text{Prob}(C_t = i) \\ &= \sum_{j=1}^{m} \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N, C_t = i, C_{t+1} = j)/\text{Prob}(C_t = i) \\ &= \sum_{j=1}^{m} \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C_t = i, C_{t+1} = j) \\ &\quad \times \text{Prob}(C_t = i, C_{t+1} = j)/\text{Prob}(C_t = i) \\ &= \sum_{j=1}^{m} \text{Prob}(\mathbf{S}_{t+1}, \ldots, \mathbf{S}_N | C_{t+1} = j) p_{ij} \\ &= \sum_{j=1}^{m} \text{Prob}(\mathbf{S}_{t+1} | C_{t+1} = j)\text{Prob}(\mathbf{S}_{t+2}, \ldots, \mathbf{S}_N | C_{t+1} = j) p_{ij} \\ &= \sum_{j=1}^{m} \pi_{\mathbf{s}_{t+1} j} \beta_{t+1}(j) p_{ij}. \end{aligned} \tag{5.10}$$

As well as calculating the likelihood, it is often of interest to determine the states of the Markov model, $\{C_t\}$, which are most likely to have generated the observed sequence. There are two possible ways of considering this:

the local problem find the local most likely state, $\bar{c}_t$:

$$\bar{c}_t = \arg \max_{1 \leq c_t \leq m} \text{Prob}(C_t = c_t | \mathbf{S}_1 = \mathbf{s}_1, \ldots, \mathbf{S}_N = \mathbf{s}_N);$$

the global problem finding the series of states $\hat{c}_1, \ldots, \hat{c}_N$ which maximises the conditional probability

$$\text{Prob}(C_1 = c_1, C_2 = c_2, \ldots, C_N = c_N | \mathbf{S}_1 = \mathbf{s}_1, \ldots, \mathbf{S}_N = \mathbf{s}_N) \,. \tag{5.11}$$

In the speech processing literature these two problems have been termed *local decoding* and *global decoding* respectively.

For the application considered later in this chapter, global decoding is appropriate, and is thus described here. It is possible to efficiently find the states $\hat{c}_1, \ldots, \hat{c}_N$ using a dynamic programming method known as the Viterbi algorithm (MacDonald and Zucchini, 1997, p. 65).

This algorithm is developed by first noting that finding the states $\hat{c}_1, \ldots, \hat{c}_N$ which maximise (5.11) is equivalent to maximising the joint probability

$$\begin{aligned} &\text{Prob}(C_1 = c_1, \ldots, C_N = c_N, \mathbf{S}_1 = \mathbf{s}_1, \ldots, \mathbf{S}_N = \mathbf{s}_N) \\ &\quad = \text{Prob}(C_1 = c_1, \ldots, C_N = c_N) \\ &\qquad \times \text{Prob}(\mathbf{S}_1 = \mathbf{s}_1, \ldots, \mathbf{S}_N = \mathbf{s}_N | C_1 = c_1, \ldots, C_N = c_N) \\ &\quad = (f_{c_1} p_{c_1 c_2} p_{c_2 c_3} \cdots p_{c_{N-1} c_N})(\pi_{\mathbf{s}_1 i_1} \cdots \pi_{\mathbf{s}_N i_N}). \end{aligned} \tag{5.12}$$

Define the quantities

$$R_{c_N}^{(N)} = \text{Prob}(\mathbf{S}_N = \mathbf{s}_N | C_N = c_N) \tag{5.13}$$

and

$$\begin{aligned} R_{c_t}^{(t)} &= \max_{c_{t+1}, \cdots, c_N} \text{Prob}(\mathbf{S}_t = \mathbf{s}_t, \ldots, \mathbf{S}_N = \mathbf{s}_N, C_{t+1} = c_{t+1}, \ldots, C_N = c_N | c_t) \\ &= \max_{c_{t+1}, \cdots, c_N} \text{Prob}(c_{t+1}, \ldots c_N | c_t) \text{Prob}(\mathbf{S}_t = \mathbf{s}_t, \ldots, \mathbf{S}_N = \mathbf{s}_N | c_t, \ldots c_N) \end{aligned} \tag{5.14}$$

where, for ease of notation, the event that $C_t = c_t$ may also be represented simply as c_t. Note that $R_{c_t}^{(t)}$ gives the partial maximisation of the probability from position t for all possible values of c_t.

The computation of (5.12) may be simplified by noting that the following recursion exists between $R_{c_t}^{(t)}$ and $R_{c_{t+1}}^{(t+1)}$:

$$R_{c_t}^{(t)} = \text{Prob}(\mathbf{S}_t = \mathbf{s}_t | C_t = c_t) \max_{c_{t+1}} \left[p_{c_t c_{t+1}} R_{c_{t+1}}^{(t+1)} \right] \tag{5.15}$$

with starting point $R_{c_N}^{(N)}$. By applying the algorithm repeatedly from $t = N-1, N-2, \ldots, 2, 1$, the quantity $R_{c_1}^{(1)}$ is obtained for all possible values of c_1. Selecting the

largest of the quantities $f_{c_1} R_{c_1}^{(1)}$ gives the relative size of the maximum probability specified in (5.12).

To carry out the global decoding step, note that, from (5.15), for each state c_t, the state $\hat{c}_{t+1}^*$ at the next position which maximises the contribution to the likelihood is known. Once the state at the first position, $\hat{c}_1$, which maximises the contribution to the likelihood is known, (5.15) gives $\hat{c}_2$, and then $\hat{c}_3$ and so on. Thus, the algorithm requires another pass through the data, this time from positions 1 to N.

Note that it is possible to carry out the algorithm in the other direction, starting the recursion at position 1 and moving forward when calculating the size of the maximal probability; details are in MacDonald and Zucchini (1997, p. 65).

The theory described above is now used in the development of a Bayesian approach to modelling recombination in phylogenetic data sets.

5.3 Modelling topology change due to recombination in a DNA alignment

Consider an alignment of T DNA sequences, each N nucleotides long. The data set may be considered as a $T \times N$ matrix, $\mathbf{S}$, with each column of the matrix, $\mathbf{S}_t$, representing the nucleotides in each sequence at a particular site. In the possible presence of recombination, the problem of estimating the phylogenetic relationships between these sequences may be viewed as that of allocating a particular topology to each position in the alignment, i.e., to each $\mathbf{S}_t$. Representing this problem in terms of the conditional independence graph in Figure 5.4, the $\{C_t\}$ correspond to the (unobservable) true topology at each site. The number of possible trees for T sequences is given by $\prod_{i=3}^{T}(2i-5)$, which rapidly increases. Thus, in the development of this theory, only data sets of four sequences are considered. In this case there are three possible unrooted trees so C_t, $t = 1, \ldots, N$ may take the values 1, 2 or 3.

As discussed previously in **5.1**, a Bayesian approach is reasonable, since the site likelihoods are readily available and it is sensible to use prior knowledge about recombination, i.e., incorporate the $\{C_t\}$ into the model. For the model to become a Hidden Markov model, the prior distribution for the $\{C_t\}$ must be the probabilities of a discrete, first-order Markov chain. Fortunately, this is a sensible choice as a first step in incorporating prior information.

5.3.1 Prior distribution for recombination events

The prior distribution for the sequence of topologies for a data set, N bp long, would specify a probability for every possible sequence of N numbers, the number at each position taking a value in $\{1, 2, \ldots, m\}$, where m is the total number of possible topologies. One way of incorporating limited dependence between the terms of this sequence is to

use a discrete-time, first-order Markov model. This is a model having the property that

$$\text{Prob}\big[N(t+1)|N(t), N(t-1), \dots\big] = \text{Prob}\big[N(t+1)|N(t)\big],$$

i.e., the state of the process at time $t+1$, $N(t+1)$, depends only on the current state of the process.

To define a Markov chain for the sequence of topologies, C_t, $1 \leq t \leq N$, the transition probabilities, p_{ij}, may be specified. This has been done as follows:

$$p_{ij} = \lambda\delta_{ij} + (1-\lambda)f_j \tag{5.16}$$

where f_j is the stationary frequency of topology j, $j = 1, 2, \dots, m$;

δ_{ij} is the Kronecker delta function (1 if $i = j$; 0 otherwise).

λ is a value between 0 and 1 representing the difficulty of changing state (topology), with a value of 0 representing an easy change of state, while a value of 1 makes it impossible to switch between states. So if $\{C_t\}$ is a Markov process, defined as above, specifying the sequence of topologies, c_t, then the prior probability of a particular sequence $c_1, \dots, c_N$ is

$$\text{Prob}(C_1 = c_1, \dots, C_N = c_N) = f_{c_1} p_{c_1 c_2} p_{c_2 c_3} \cdots p_{c_{N-1} c_N}.$$

Choosing a prior is quite subjective as it is difficult to select a vague prior. The prior may be uninformative in that the stationary frequencies of all the possible topologies can be assumed to be all equal. However, the value of λ must also be specified and this may introduce a degree of subjectivity. Therefore, an investigation of the sensitivity of the results to the choice of prior will be carried out later (see **5.4**).

5.3.2 Likelihood

Superficially, the problem of calculating the likelihood seems straightforward: for each possible topology, calculate the likelihoods for each $\mathbf{S}_t$ (i.e., each column of the alignment). Then, for a particular sequence of site topologies for $\mathbf{S}_1 \dots \mathbf{S}_N$, the corresponding likelihoods may be multiplied together to yield the overall likelihood. However, upon closer consideration, a possible problem arises – the branch lengths.

The branch lengths can have a considerable effect on the probability of observing a particular pattern of nucleotides at a site for a given topology. Therefore, choosing reasonable values is important. An obvious way to estimate the branch lengths might be to simply maximise the likelihood over the branch lengths for each of the possible topologies for a given data set. Unfortunately this approach is potentially flawed: if one or more recombination events have occurred, then the estimation of the branch lengths will be inaccurate.

To explain this, suppose that, in a DNA data set, one recombination event has occurred, with the affected subset of the DNA sequence being considerably shorter than the entire sequence length. There are two topologies valid for this data set: top_0, the topology in the nonrecombinant regions and top_1 in the recombinant area. Since the recombinant region is small, estimation of the branch lengths for top_0 based on the entire sequence will not be too adversely affected by the conflicting signal coming from the recombinant zone. However, when estimating the branch lengths for top_1, the valid signal from the recombinant part of the sequence will be sometimes swamped by the misleading information coming from the rest of the sequence where top_1 is incorrect. If the branch length values for top_1 are wrong, then the method may lose power. Thus, some form of localised calculation of the likelihoods may be a solution.

To calculate likelihoods locally, the sequence may be split into subsets and likelihoods calculated on each. The issue is what size of subsets to use. There are two extremes: subsets large enough to run into the problem described above; and subsets of one column of the alignment. This latter approach will not have problems of conflicting phylogenetic signal, but it will throw away a lot of the information on branch lengths contained in neighbouring sites. This increases the variance of the branch length estimates and again, any method using these likelihoods will lose power.

So there are two opposing effects: the conflicting phylogenetic signal coming from large subsets containing two or more different topologies (and other heterogeneities such as substitution rate variation for many real data sets) and the increased variance of the branch length estimates when small, homogeneous subsets of the data are used. It is possible that one effect may dominate the other, leading to large or small subsets being used, or a trade-off between the two may be necessary. This point is considered in **5.4.1** and in **5.5**.

Once the decision on the size of the subset has been made, the likelihood values

$$\text{Prob}(\mathbf{S}_j = \mathbf{s}_j | C_j = c_j)$$

should be calculated for all $\mathbf{S}_j$, $j = 1, \ldots, N$ and for all m topologies.

5.3.3 Posterior distribution

Now that the prior distribution and the likelihood have been specified, the relationship

$$\textit{posterior} \propto \textit{prior} \times \textit{likelihood}$$

may be used to find the posterior distribution. Substituting in the prior and the likelihood, the posterior is given by

$$\begin{aligned}\text{Prob}(c_1, c_2, \ldots, c_N|\mathbf{S}^{(N)}) &\propto \text{Prob}(c_1, \ldots, c_N)\text{Prob}(\mathbf{S}^{(N)}|c_1, \ldots, c_N) \\ &= f_{c_1} p_{c_1 c_2} \cdots p_{c_{N-1} c_N} \prod_{j=1}^{N} \text{Prob}(\mathbf{S}_j = \mathbf{s}_j | C_j = c_j) \quad (5.17)\end{aligned}$$

where $\mathbf{S}^{(N)} = (\mathbf{S}_1, \ldots, \mathbf{S}_N)$. Clearly, this posterior distribution is formulated as a Hidden Markov model, as described in **5.2.1**. The calculations described in **5.2.3** may be carried out for this model. In particular, the sum of all the terms (the renormalisation constant) may be calculated using the forward and backward probabilities, while the most probable combination of topologies may be found using the Viterbi algorithm.

To calculate the renormalisation constant, (5.6) may be used. In the C programs written to carry out these calculations, t in this equation has been set to one. Hence, the forward probabilities $\alpha_1(j)$ and the backward probabilities $\beta_1(j)$ must be calculated for $j = 1, \ldots, m$. The former are simple to find; (5.8) gives the formula for calculating these forward probabilities. To compute the backward probabilities, the recursion given in (5.10) must be used. The recursion may be started by noting that $\beta_{N-1}(j) = \text{Prob}(\mathbf{S}_N = \mathbf{s}_N | C_{N-1} = j)$ since $\beta_N(j) = 1$ for all possible values of j.

Since this approach is Bayesian, the intuitive estimate of recombination events is the *maximum a posteriori* (MAP) estimate. This is simply the sequence of topologies which maximises the probability in (5.17), i.e., the solution to the global decoding problem discussed in **5.2.3** and is found using the Viterbi algorithm.

5.4 Performance of this model

It is not enough to describe a model for topology change along an alignment; the model must be tested to see if it can yield useful inferences about the presence of recombination in a phylogenetic data set. Therefore, a small simulation study has been carried out to investigate the performance of this model. Since this is a Bayesian approach, it is also important to carry out an investigation into the sensitivity of the results to the choice of prior distribution. So, for a variety of recombination events, the prior is varied and the results are compared. This achieves the dual purpose of evaluating the method and testing the importance of the choice of prior.

Before this investigation may be carried out, however, the subset size for calculating the likelihoods at each column must be chosen. The dependence of the results of the subset size was explored for a number of data sets and prior distributions.

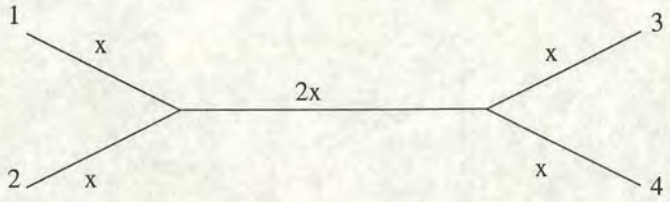


Figure 5.5: The tree used to simulate recombinant data sets. The length of each of the exterior branches is x while the length of the interior branch is $2x$.

5.4.1 The effect of the sequence subset size on likelihood calculations

It is possible that the size of the subset used to estimate the branch lengths in the site likelihood calculations plays an important role in the performance of the method. Accurate branch lengths will generally yield better likelihood values for the true tree in a particular region, and lower values for the incorrect topologies, thus playing a part in determining the power of this procedure. To improve the estimation of the branch lengths, the amount of data involved in the estimation should be as large as possible, for example, the entire data set. However, if a recombination event is short relative to the entire sequence length, then the amount of correct signal for the branch lengths of the topology resulting from the recombination will be small relative to the incorrect information from the rest of the sequence. This suggests using smaller subset sizes to calculate the branch lengths, although this will lead to increased variances of the branch length estimation. The question is whether a trade-off between these two phenomena is required, or whether one of these effects dominates the other.

To investigate this, various data sets were examined. Two are reported here. The data sets were simulated using the tree in Figure 5.5. The data sets consisted of four sequences, each containing 1000 nucleotides, related in the non-recombinant region by the tree shown in Figure 5.5. The value of x was chosen to be 0.2 substitutions per position (a typical branch length). To generate the recombination event, the four sequences were evolved along the interior and then the exterior branches using the Kimura two Parameter model of nucleotide substitution with a transition-transversion ratio of 2 (see **2.5.2**), until their length was $0.25x$ or $0.75x$. At this point the subsequence from 351 to 450 nucleotides in sequence 3 replaced the corresponding subsequence in sequence 1. The sequences then continued to evolve along the exterior branchs for the remaining length. Thus, two data sets with short recombination events have been generated, with one happening more distantly in time than the other. These recombination events should be relatively difficult to detect.

For each of the data sets, eight different subset sizes, ranging from 1 to 1000 were used to calculate the likelihoods. Six different prior distributions were used, corresponding to λ from (5.16) taking the values 0.5, $0.\dot{6}$, 0.75, 0.8, 0.9 and 0.999. The three

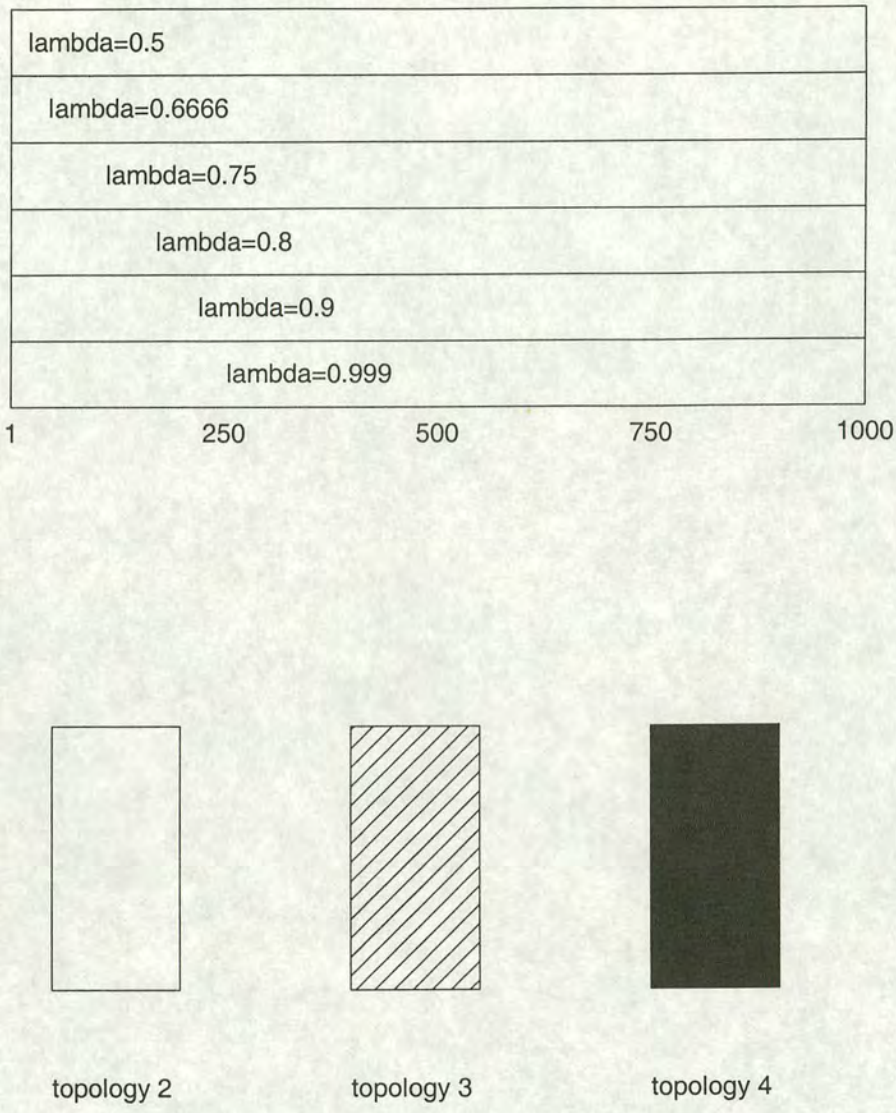


Figure 5.6: Key to the graphs in this section. The graph shows the horizontal axis, depicting location along the 1000 bp sequence, while the vertical bars correspond to particular values of λ. Below the different shadings, corresponding to each of the three topologies are shown.

stationary frequencies were all equal ($f_i = 1/3,\ i = 1, 2, 3$). The results are shown in Figures 5.7 and 5.8. A key to the graphs is given in Figure 5.6.

In each of the graphs shown in Figures 5.7 and 5.8, the horizontal axis represents the sequence of nucleotides from 1 to 1000 bp. The different shadings (none, hatched or solid) correspond to the three topologies (labelled topologies 2, 3 and 4 to represent that sequence 1 clusters with sequence 2, 3 or 4 respectively), as shown in Figure 5.6. The results from the six priors, corresponding to the six different values of λ are presented in sequential order, with the uppermost line being the prior with λ taking the value 0.5 and the lowest line corresponds to the prior with $\lambda = 0.999$. The dotted lines in Figures 5.7 and 5.8 represent where the recombinant region lies in each data set; the

absence of these lines (due to a change in topologies) means that the detected start or end of the recombinant region coincides with the actual location. In the recombinant region, the true topology is 3 (represented by hatching). Elsewhere, topology 2 applies (no shading). Topology 4 (solid filling) should not be observed.

For the data set with the relatively recent recombination event (the event occurs three quarters of the way along the exterior branches), the choice of subset size does not appear to greatly affect the MAP estimate. The recombination event is found for any subset size and for most of the prior distributions. For this data set, the MAP estimates using a subset size of 1 or 1000 seem to be the best.

The subset size does affect the results from the data set containing the more distant recombination event. For small subset sizes (1–50 bp), this recombination event is not detected at all. For the larger subset sizes, the event is found. This suggests that the reduced variance of the branch length estimates which results from using more data outweighs the conflicting phylogenetic signal in this data set.

Note that while the location of the recombination event is reasonably estimated when the subset size is 200 nucleotides, the resulting topology estimated is incorrect. This is not the case for other subset sizes where the MAP estimate finds the recombination event. This illustrates a further effect that the choice of subset size could have.

One final point to note is that the data sets examined here are homogeneous apart from the recombination event. In each data set, the branch lengths have similar lengths, the same model of nucleotide substitution is valid throughout and there is no substitution rate variation along the sequence. Real data sets are likely to be quite heterogeneous so it is possible that the conflicting phylogenetic signal could have a larger effect in practice. This point is returned to in **5.5**.

The speed of the computer program written to implement these calculations appears to depend on the subset size for calculating the likelihoods only as the time to find the MAP estimate seems negligible. For the data sets used above, the larger subset sizes gave good results. Since the data sets used in the simulation study below are simulated in a similar manner, it seems reasonable to use the largest possible subset size (the entire sequence length, 1000 bp in this example) to calculate the site likelihoods.

5.4.2 Sensitivity to the choice of a prior distribution

To test the sensitivity of the results to the choice of the prior distribution, various recombination events in data sets were simulated. The tree used to simulate the data is that shown in Figure 5.5. The outer branch lengths are x while the interior branch length is $2x$. For the simulation study here, x takes on two values: 0.05 and 0.2 substitutions per position. The data were simulated along the interior branch and

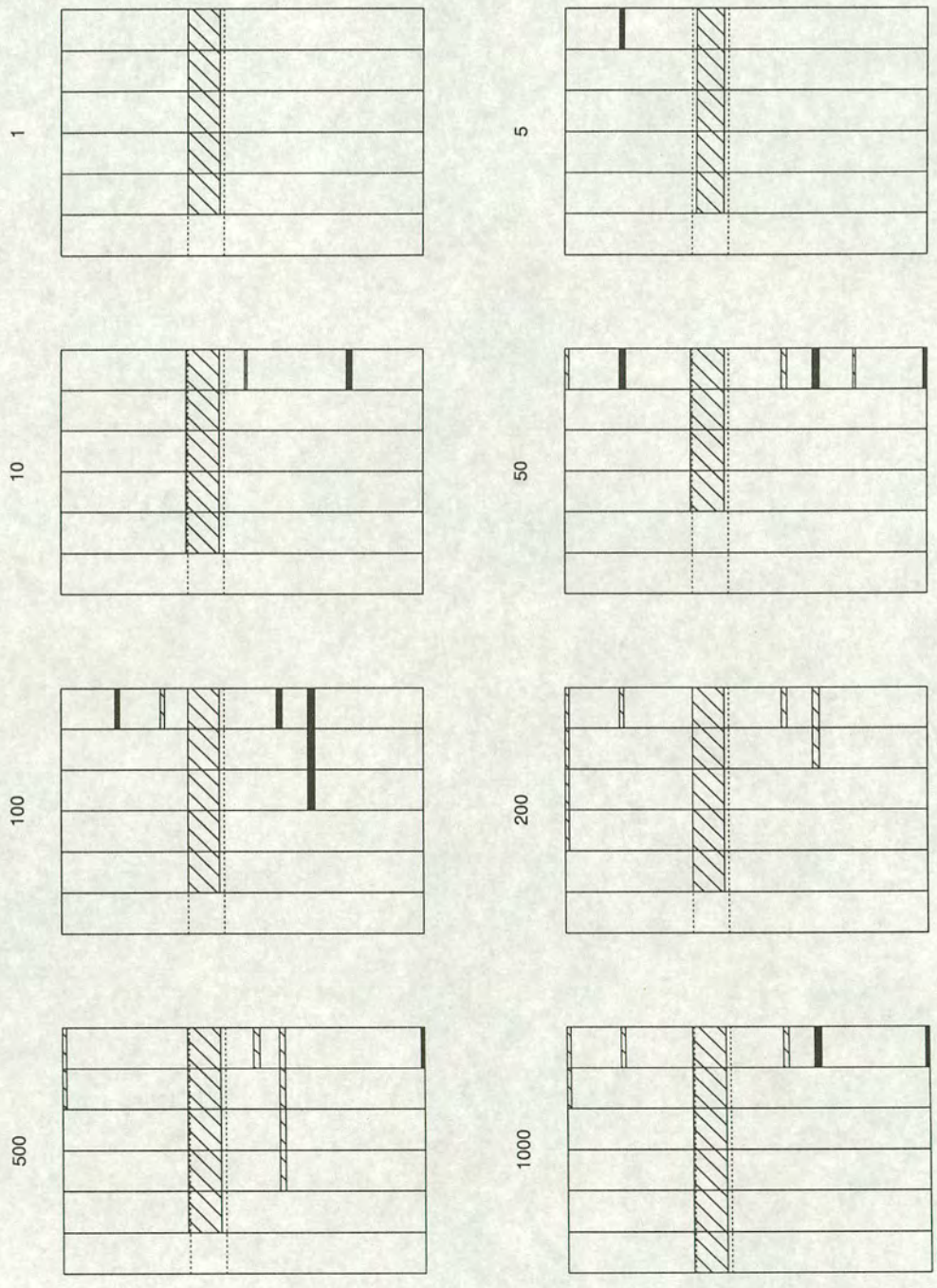


Figure 5.7: The effect of different branch lengths for a tree with a recombination event between 351 and 450 bp. $x = 0.2$ and recombination occurs when the exterior branches have attained 3/4 of their length. The subset sizes used to calculate the likelihood are shown on the left of the graphs.

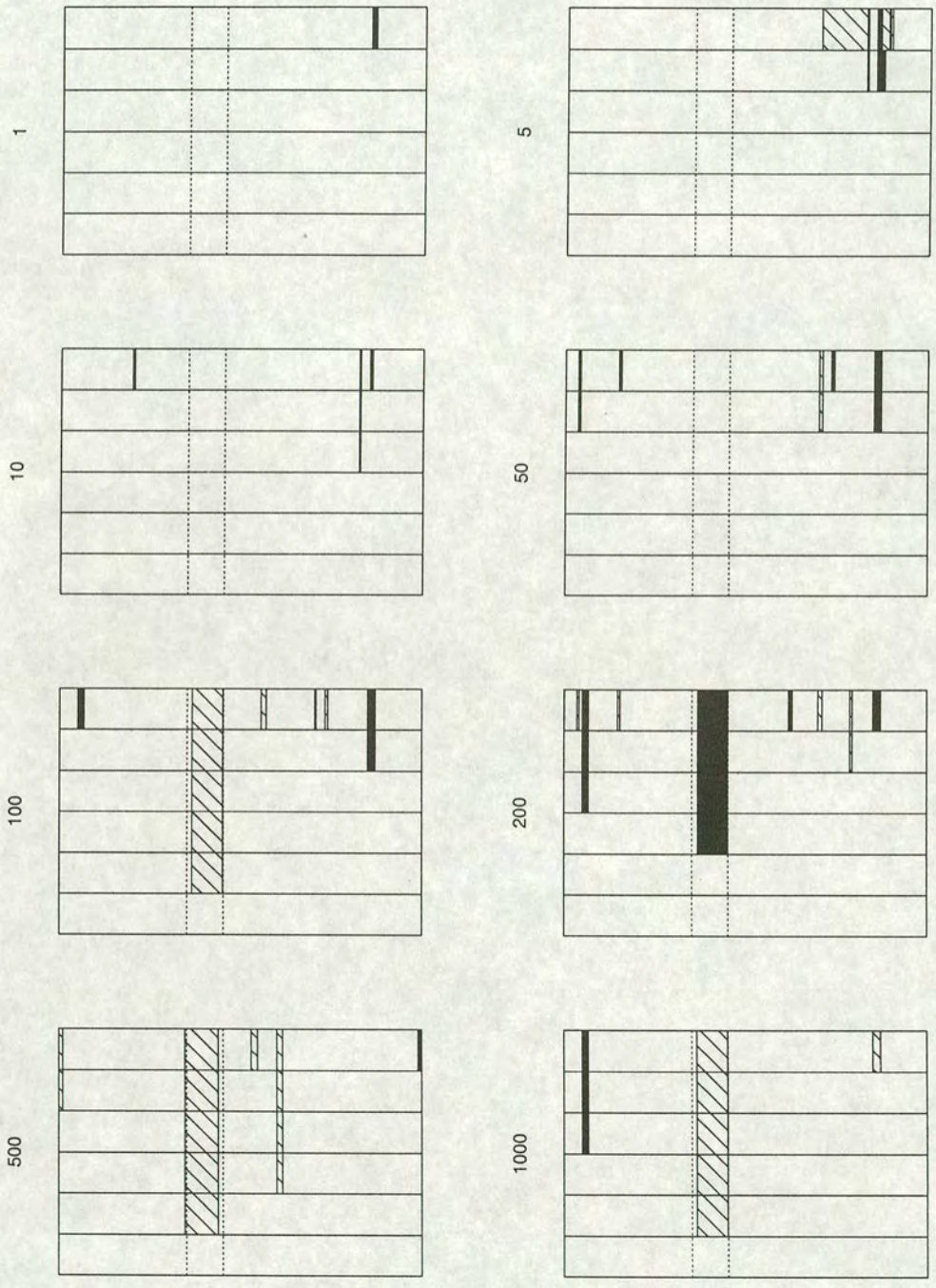


Figure 5.8: The effect of different branch lengths for a tree with a recombination event between 351 and 450 bp. $x = 0.2$ and recombination occurs when the exterior branches have attained 1/4 of their length. The subset sizes used to calculate the likelihood are shown on the left of the graphs.

along the four outer branches until their lengths were a fraction, b, of the total branch length. Then a recombination event was generated, with a region of sequence 3 replacing the same region in sequence 1. Following that, the sequences were evolved along the remainder of the length of the branches ($[1-b]x$ substitutions per position). Values of b were 0.25 and 0.75.

The data were simulated using a Kimura two Parameter model (the transition-transversion ratio was chosen to be 2). The sequences were 1000 bp long, with three lengths of recombination event: 400 (positions 301–700); 200 (positions 301–500) and 100 (positions 351–450). The subset length for calculating the likelihoods was 1000. The prior distributions used were all similar in that the stationary frequencies of each of the three topologies were all equal (to 1/3). On the other hand, the value of λ, from (5.16), was varied, taking on six possible values: $\lambda = 0.5, 0.\dot{6}, 0.75, 0.8, 0.9, 0.999$. For each set of conditions, five data sets were simulated. This should represent, to some extent, the possible range of results. The data were then analysed using the Bayesian model described above. Recombination events were inferred using the MAP estimate. The results are shown in graph form in Figures 5.9 to 5.14. The information represented in the graphs has been explained above, and in Figure 5.6.

Various conclusions may be drawn from the results of this simulation study. Firstly, the degree of 'patchiness' (the presence of short switches in topology) of the results decreases as the value of λ increases. This is not surprising since λ reflects the difficulty in changing topology; the higher the value, the less worthwhile it is for the topology to switch, despite the presence of higher likelihood values. As λ gets very large (≥ 0.9), the MAP estimate for some of these data sets suggests that no recombination event has occurred. Again this makes sense: high values of λ require a lot of support for a change of topology from the site likelihood values before a recombination event is inferred. The site likelihoods for old, short recombination events may not be high enough to cause a change in topology when λ is high.

The most difficult event to detect is the short recombination (100 bp long) in the short tree ($x = 0.05$) which occurs early in the evolution of the data set ($b = 0.25$). This is not surprising since this is a distant recombination event which occurs between relatively closely related sequences. The fact that a recombination event is sometimes estimated is promising.

The depth in the tree at which a recombination event occurs is an important factor in determining the difficulty of the estimation problem. This is obvious: if a recombination event occurs far back in time then more of the signal from the event will be overwritten by nucleotide substitutions occurring afterwards than for a more recent event. When $b = 0.75$ (a recent event), the results from both trees and for all lengths of event are generally good - for most values of λ the recombination event is detected to a reasonable

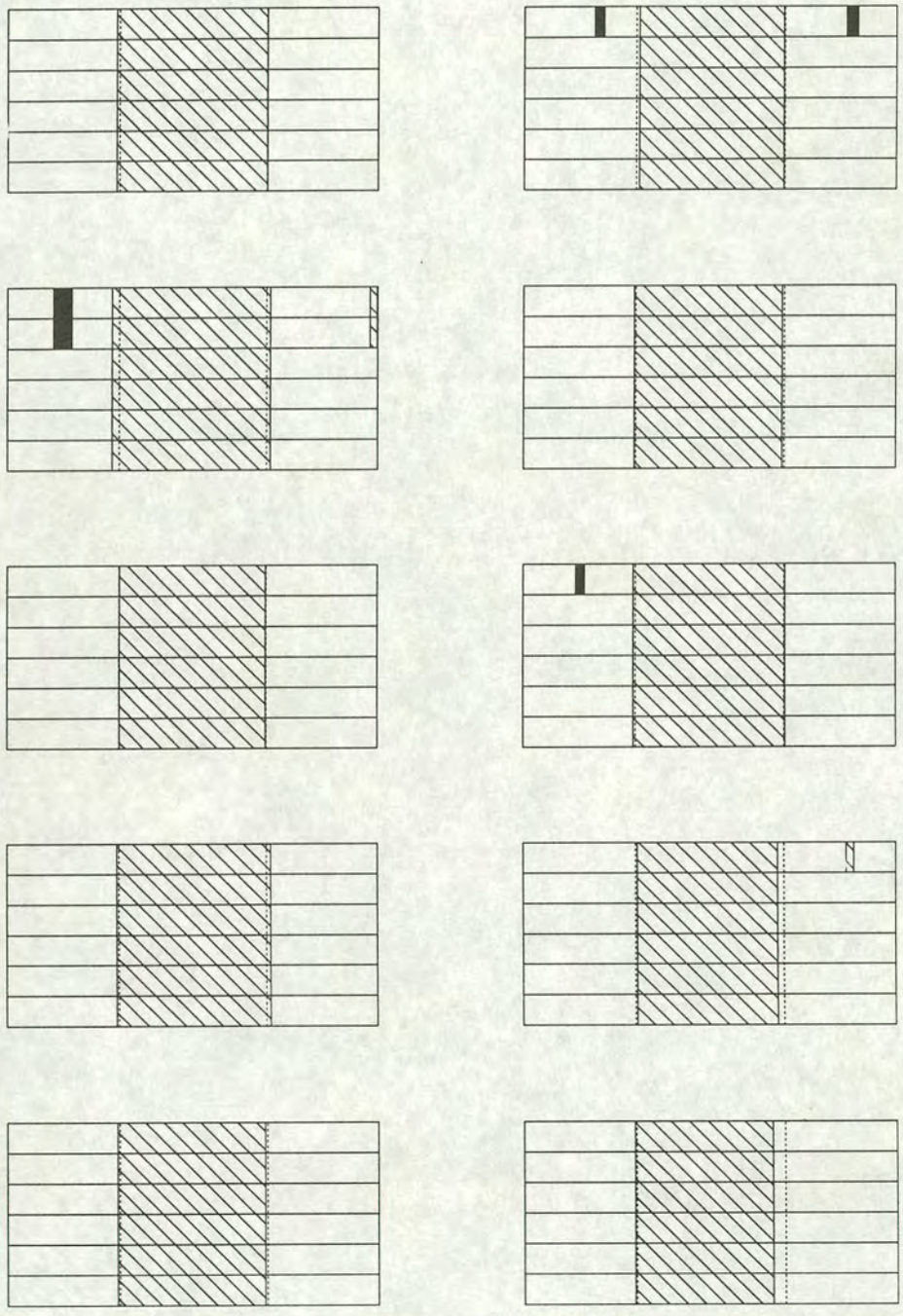


Figure 5.9: A recent recombination event ($b = 0.75$), 400 bp long, occurring between the dotted lines (301–700 bp). x (from Figure 5.5) is 0.05 for the left-hand graphs and $x = 0.2$ for the right-hand graphs. Five data sets were simulated for each value of x.

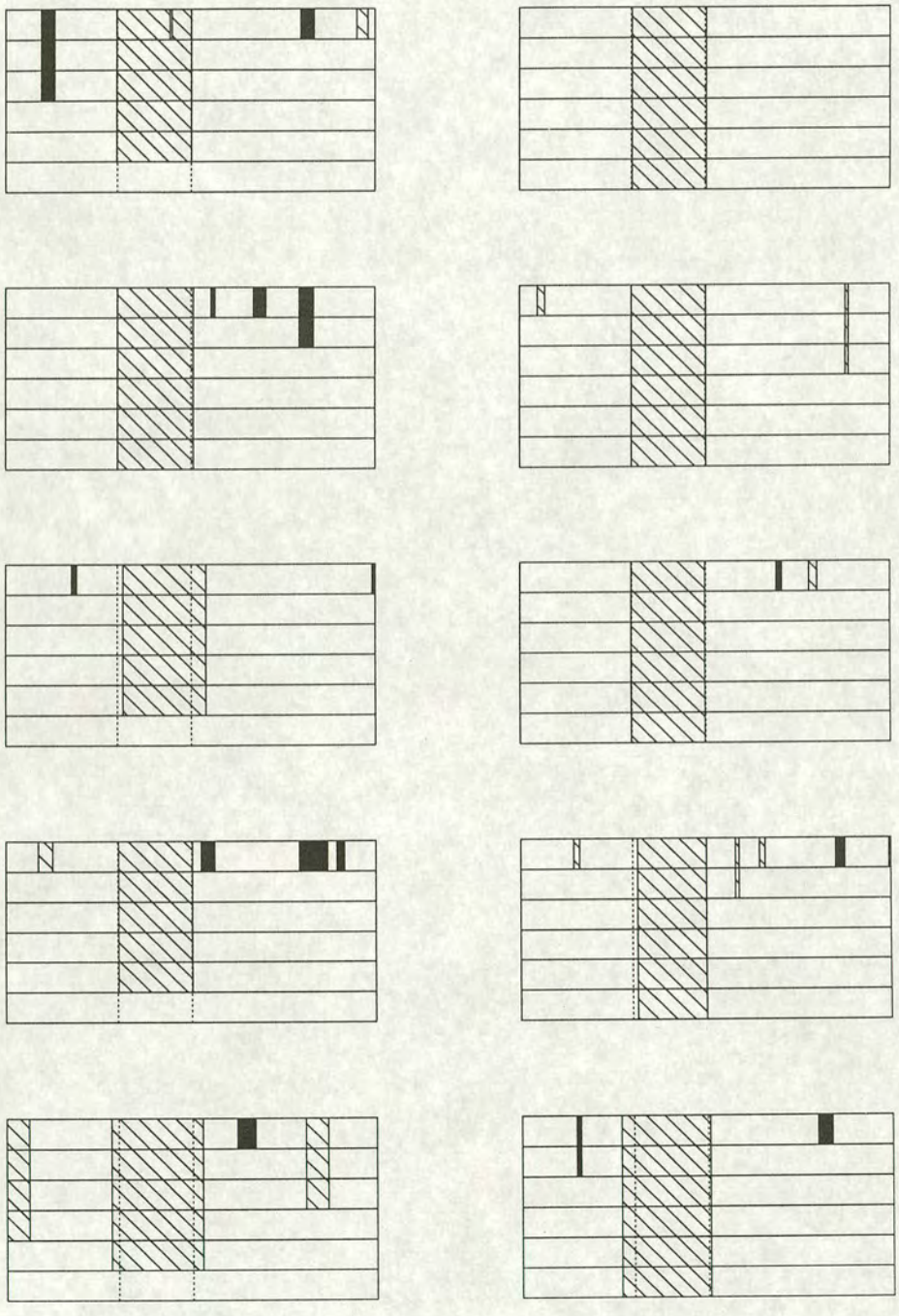


Figure 5.10: A recent recombination event ($b = 0.75$), 200 bp long, occurring between the dotted lines (301–500 bp). $x = 0.05$ for the left-hand graphs and $x = 0.2$ for the right-hand graphs. Five data sets were simulated for each value of x.

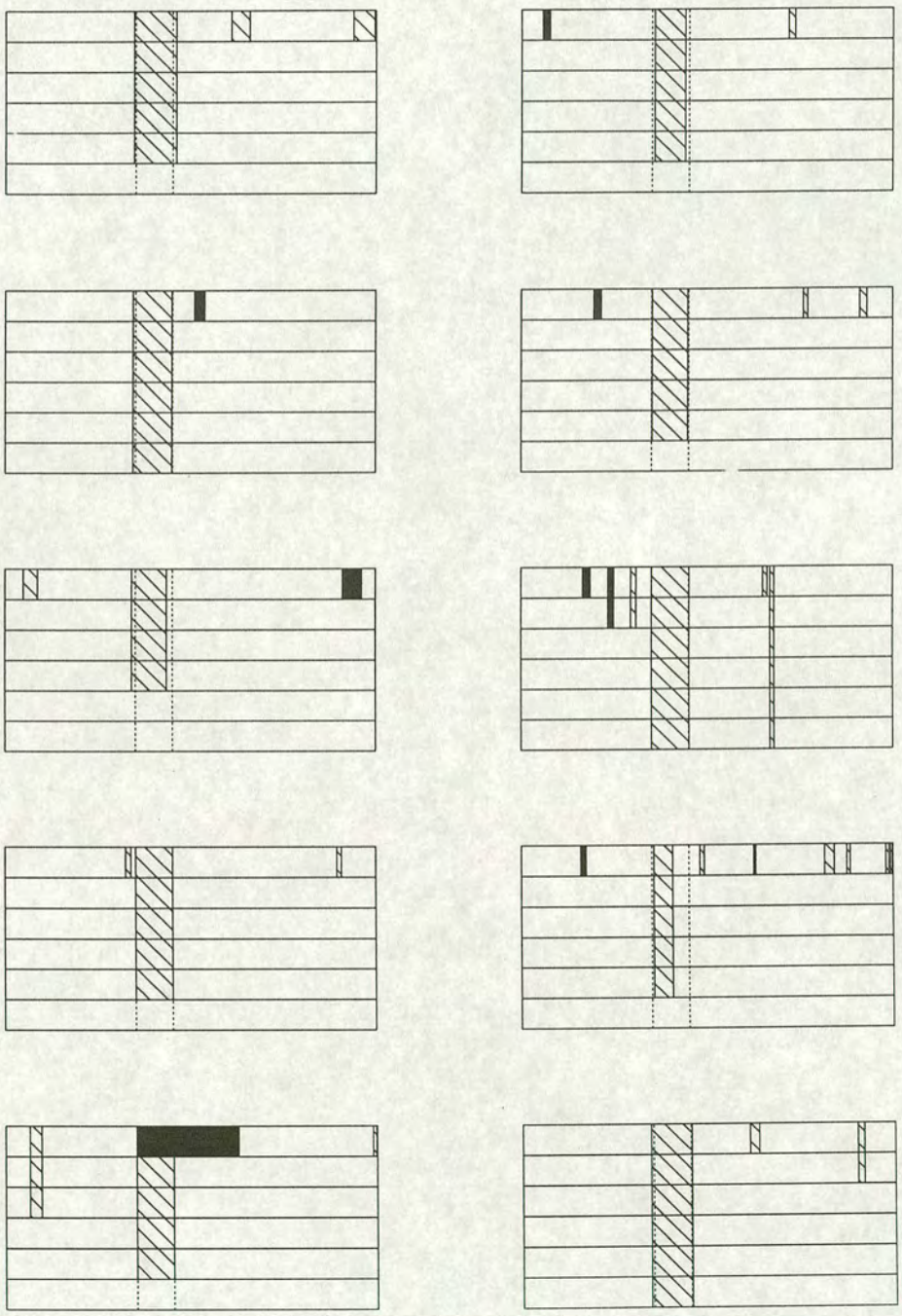


Figure 5.11: A recent recombination event ($b = 0.75$), 100 bp long, occurring between the dotted lines (351–450 bp). $x = 0.05$ for the left-hand graphs and $x = 0.2$ for the right-hand graphs. Five data sets were simulated for each value of x.

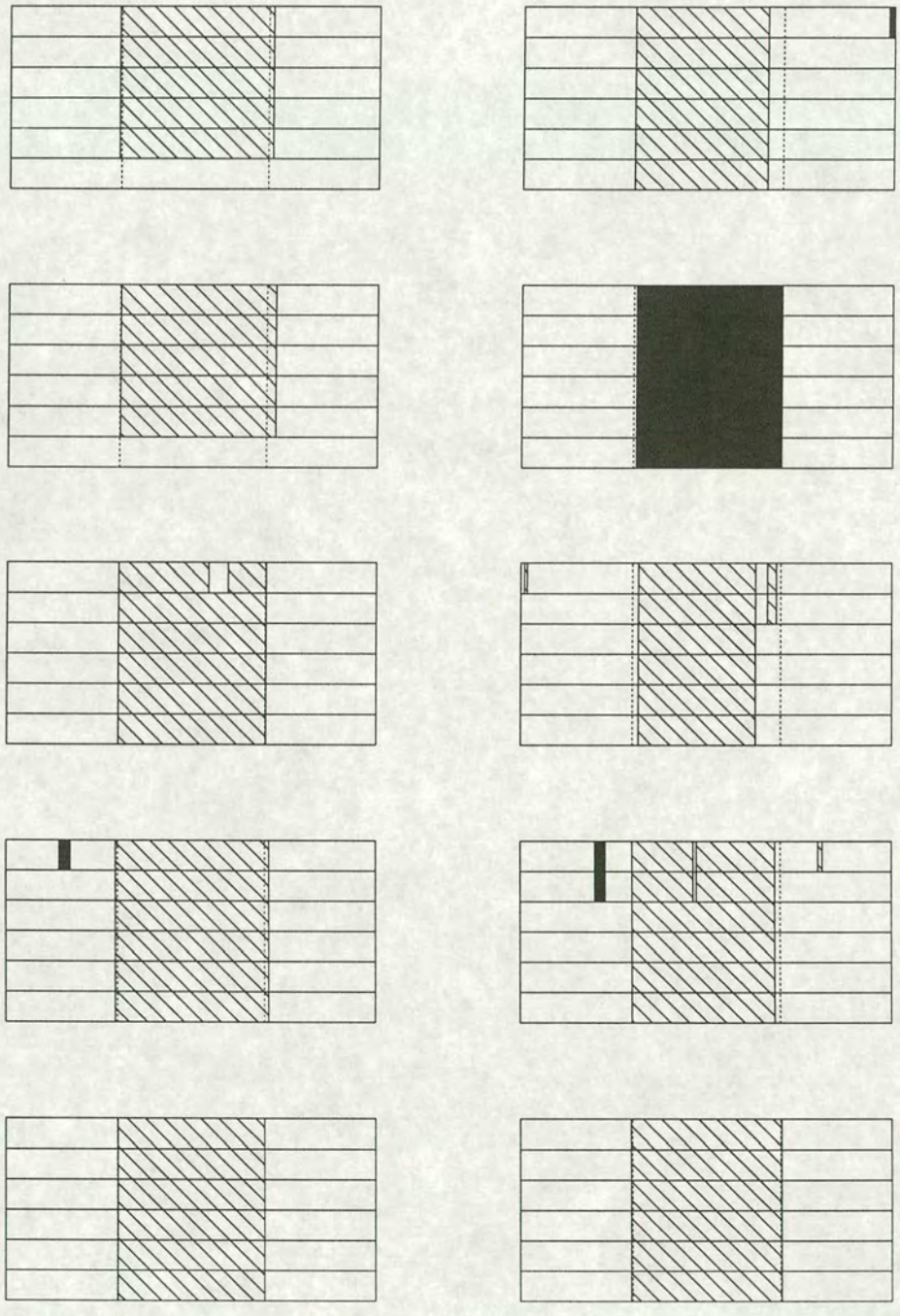


Figure 5.12: A distant recombination event ($b = 0.25$), 400 bp long, occurring between the dotted lines (301–700 bp). $x = 0.05$ for the left-hand graphs and $x = 0.2$ for the right-hand graphs. Five data sets were simulated for each value of x.

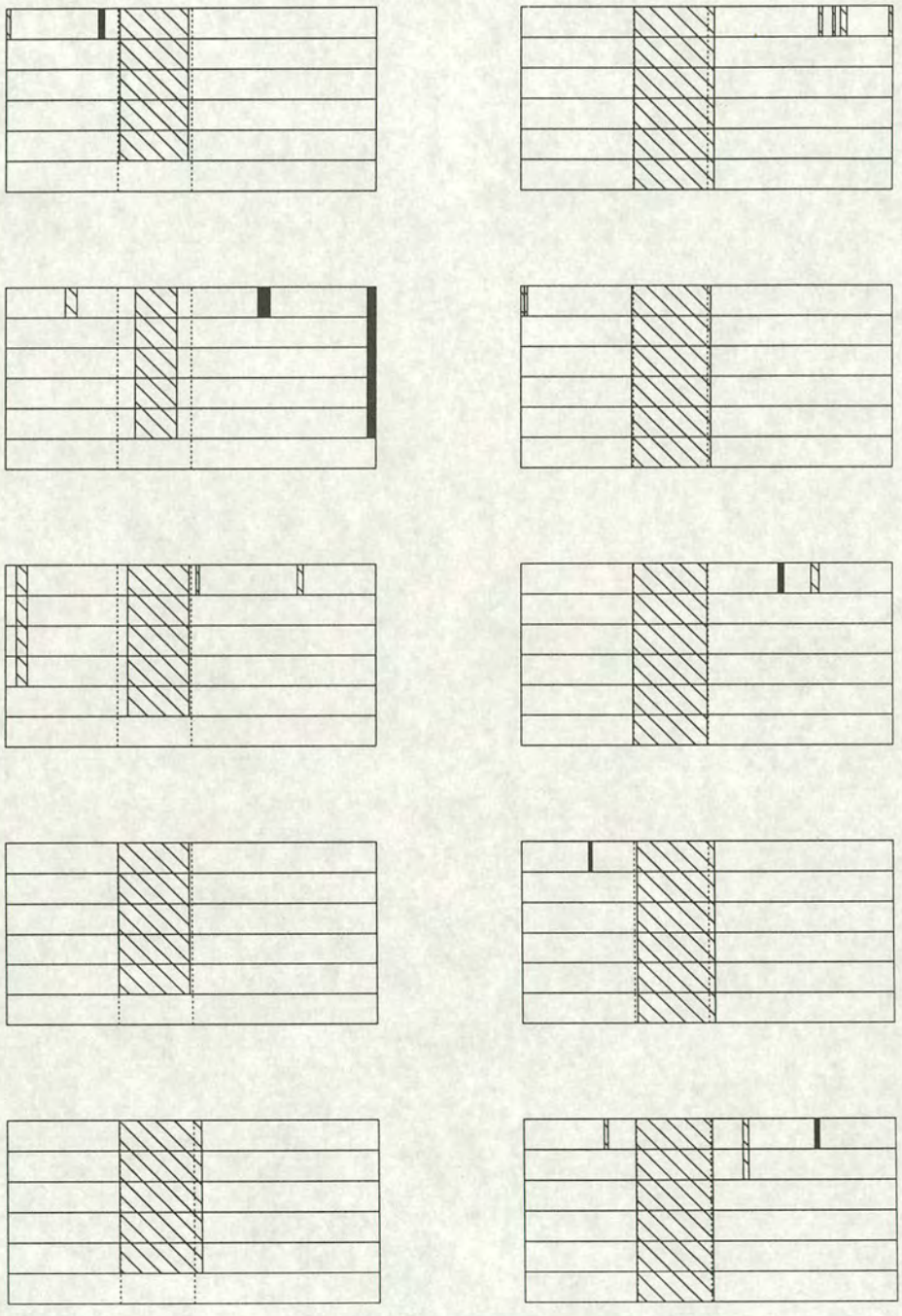


Figure 5.13: A distant recombination event ($b = 0.25$), 200 bp long, occurring between the dotted lines (301–500 bp). $x = 0.05$ for the left-hand graphs and $x = 0.2$ for the right-hand graphs. Five data sets were simulated for each value of x.

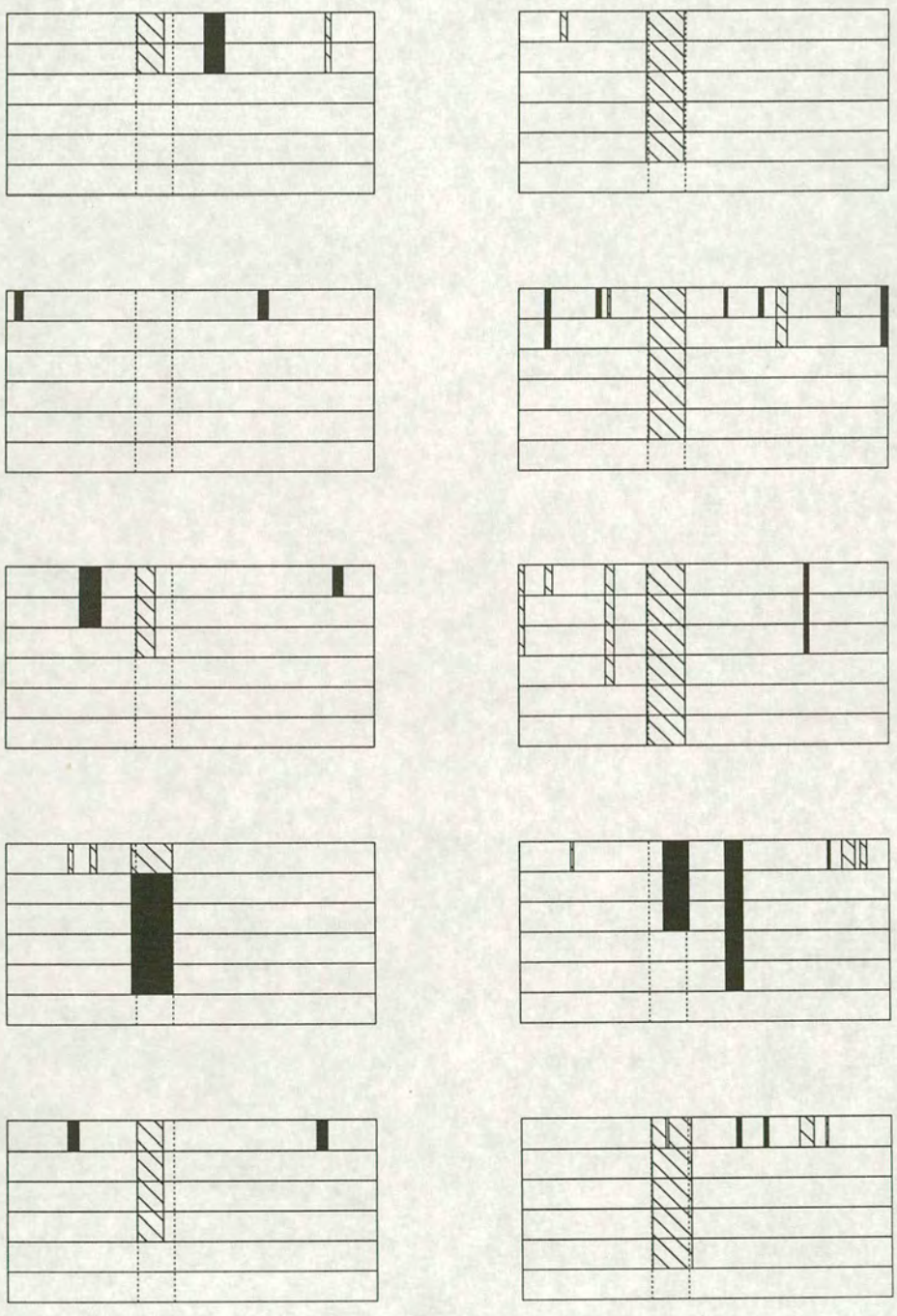


Figure 5.14: A distant recombination event ($b = 0.25$), 100 bp long, occurring between the dotted lines (351–450 bp). $x = 0.05$ for the left-hand graphs and $x = 0.2$ for the right-hand graphs. Five data sets were simulated for each value of x.

degree of accuracy. However, looking at events further back in time ($b = 0.25$), the performance of the method diminishes. This is not so noticeable for long recombinant regions but is very apparent for the shortest recombination event simulated (100bp); the MAP estimate for some of these data sets does not infer a recombination event.

Another observation which stems from the above is that the success at detecting recombination depends on the length of the region. Obviously a larger recombinant region is easier to detect since a larger set of site likelihood values in the sequence support the topology resulting from the recombination event. For example, in the ideal scenario the site likelihoods for data sets with the shortest recombination region (100 bp) should be higher for the recombinant topology at those sites in the 10% of the sequence where the recombination has occurred. This is a relatively small proportion of the sequence affected by the recombination event (in comparison with 40%). In addition, random mutations will obscure some of the signal, which may mean that other topologies are favoured at some of the sites. This leads to a reduction in information, and therefore, it becomes harder for the model to detect recombination.

In some data sets, the location of the recombinant region is correctly inferred, but the topology is not. Examples of this are the fourth data sets for the two sets of conditions for the distant recombination event of length 100 bp (see Figure 5.14). As mentioned in **5.4.1**, a different choice of subset size for the likelihood calculations might change matters. The site likelihoods were calculated using subset sizes of 500 bp and 200 bp and for both of them, a subset size of 200 bp led to not only the correct location being inferred but also to the correct topology (results not shown). The fact that this problem has occurred in this small simulation study suggests that a more detailed investigation of the dependence of the results on the subset size must be carried out.

Overall, from this simulation study, it does appear that the model of recombination proposed above works quite well. Depending on prior beliefs about the recombination event, an appropriate value of λ can be selected (if it is believed that a putative recombination event is short or occurred quite far back in the tree, a lower value of λ should be selected). From the simulation study, the results appeared quite stable over a wide range of values of λ, apart from the patchy effect. This suggests that, in any analysis, a range of values of λ should be used. Putative recombination events which persist across these values are more likely to represent an actual recombination event rather than being an artifact of the data.

5.5 Example using a Neisseria data set

The model described above is now applied to a real data set, with a known recombination event. The data set used is a subset of the *Neisseria* sequence data for the *argF* gene used in Chapter 4. The complete data set had eight strains of *Neisseria*; the

Table 5.2: MAP estimates of recombination events for the *Neisseria* data set

λ	0.5	$0.\dot{6}$	0.75	0.8	0.9	0.999
	296–342(3)[a]					
	357–498(3)	296–498(3)	296–498(3)	296–498(3)	296–498(3)	296–498(3)
	827–864 (2)					

[a]unspecified sites have topology 1

data set used here consists of four of these strains: *N. gonorrhoeae* (accession number X64860); *N. meningitidis* (X64866); *N. cinerea* (X64869) and *N. mucosa* (X64873). Further details on these sequences are available in Zhou and Spratt (1992). The alignment of these sequences was carried out using CLUSTAL W (Thompson et al., 1994), taking the default settings. The alignment is 787 bp long. Following the number scheme of Zhou and Spratt (1992), the first nucleotide is labelled as 296 bp with the last one at 1082 bp.

According to Zhou and Spratt (1992), there are two anomalous, or more diverged regions in the DNA alignment. These occur at positions 296–497 bp and 802–833 bp. In the rest of the sequence, *N. meningitidis* clusters with *N. gonorrhoeae* (later referred to as topology 1) while between 296 bp and 497 bp they found that it is grouped with *N. cinerea* (topology 3). Zhou and Spratt (1992) were not able to determine the cause of the other diverged region (802–833 bp).

Before applying the model various parameters must be estimated. From the data, the equilibrium frequencies, π_i, $i = A, C, G, T$ of the four nucleotides were estimated as $\pi_A = 0.26$, $\pi_C = 0.28$, $\pi_G = 0.28$ and $\pi_T = 0.18$. Using the PUZZLE program (Strimmer and von Haeseler, 1996), the transition-transversion ratio was estimated as 2.3. In keeping with earlier remarks about trying different values of λ, six different values were used ($\lambda = 0.5$, $0.\dot{6}$, 0.75, 0.8, 0.9 and 0.999 – those values which were used in the simulation study).

The final question concerns the subset size to use to calculate the likelihoods. Various subset sizes were used and it was found that, for this data set, the problems of conflicting phylogenetic signal outweighed the effects of increased variance in the branch length estimates. Using the entire sequence to find the branch lengths resulted in the incorrect identification of a recombination event whereas the recombination event was correctly located and identified using a very small subset size (5 nucleotides). The results, using this subset size, for the six different prior distributions are shown in Table 5.2.

Apart from the patchiness in the results when $\lambda = 0.5$, the method finds the larger recombination region successfully over a wide range of values of λ. It also correctly identifies the change in topology, with the sequence of topologies at each site starting with topology 3, then changing to topology 1. The method is not successful at identify-

ing the shorter diverged region. This is not surprising as Zhou and Spratt (1992) were unable to determine the cause of this diverged region; if it is a recombination event, the recombinant DNA does not appear to originate from any of the strains in their data set. There is a change in topology towards the end of this diverged region when $\lambda = 0.5$. This may be picking up genuine information in the data, or it may be an artifact due to the low value of λ. Since it does not persist for some of the higher values of λ, the reasonable conclusion would be to ignore it.

If the entire sequence length is used to find the site likelihoods, a recombination event is estimated between 296 and 829 bp for high values of λ. This incorrect estimate probably results from the heterogeneities in the data. In the simulation study described in **5.4.2**, small subset sizes (10 bp) performed almost as well as using the entire data set. Since real data sets are often heterogeneous, unlike those in the simulation, it is possible that small subset sizes are optimal in practice. This point requires further investigation.

5.6 Discussion and future work

The Bayesian approach to detecting recombination described in this chapter follows naturally from considering the problem in the framework of a Hidden Markov model. However, the structure of this approach is also very similar to the parsimony-based method suggested by Hein (1993). This procedure for detecting recombination has previously been described in Chapter 3 (see **3.2**). Hein also considers the problem in terms of a graph, containing N nodes, each linked to the one directly preceding it. Each node t is assigned a weight, $w(t, c_t)$, the weight of position t given it has topology c_t. In the Bayesian approach described here, this corresponds $\text{Prob}(\mathbf{S}_t = \mathbf{s}_t | C_t = c_t)$, the site likelihood, given topology c_t. The edge connecting nodes t and $t-1$ is assigned a weight, $d(c_t, c_{t-1})$, the recombinational distance between topologies c_t and c_{t-1}. This is equivalent to the transition probabilities given in (5.16). The estimate of the location (and consequences) of the recombination events in the data set is the most parsimonious path through this graph while for the Bayesian approach, it is the path of highest probability.

Due to this correspondence, it should be possible to incorporate some of this methodology to extend the application of this work. For small data sets (≤ 6 sequences), he considers the possible topologies that could arise from the current topology through one or more recombination events. This restricts the number of possible topologies that need to be considered at each node. The same rules could be used in the Bayesian approach to extend the method to data sets of 5 or 6 sequences.

For larger data sets, he describes a heuristic method which overcomes the high computational burden of employing the exact approach for large data sets. Essentially, this

assumes that firstly, the topology at one point in the sequence is known and secondly, that only one recombination event may occur between each node (nucleotide). This reduces the number of topologies that need to be considered. Again, these ideas could be used in the Bayesian approach.

Extending the Bayesian approach in this way could prove computationally tedious, since at each node, the site likelihoods for the permissible topologies would need to be calculated. A quicker approximate approach might be to use the idea of quartets; this was suggested by Strimmer and von Haeseler (1996) to approximate a maximum likelihood tree. A large data set could be split into quartets of four sequences (not all quartets would need to be examined) and each quartet could be analysed as described above. Many of these would contain no recombination event and thus could be ignored. Others might find evidence of a recombination event. The results from such quartets could be combined at the end and an overall estimate of recombination could be obtained for the entire data set. This procedure would not be trivial to implement and would require further attention to assess its validity.

One obvious point which should also be addressed is the selection of the value λ. From a practical viewpoint, it might be known from other studies that certain parts of particular sequences have low levels of recombination whereas other regions may be more likely to contain recombinant regions. Appropriate values of λ could be defined in these regions (e.g., $\lambda = 1$ if recombination is impossible). This could be easily implemented.

It could be argued that the best way to incorporate changing values of λ and/or remove the subjectivity in the choice of prior is to place a hyper-prior on λ. Thus, as well as the topology categories, a value for λ would have to be estimated. Ignoring the computational difficulties for the present (a hyper-prior on λ might cause the Hidden Markov model structure to fail), it is unclear whether such an approach is valid.

To explain this, consider the maximisation of the posterior probability (5.17) over λ (this is equivalent to putting a uniform hyper-prior on λ). So the object is to find the combination of topologies and the value of λ which maximises the posterior probability. To investigate the consequences of this approach, three data sets, with different recombination events, were generated as described in **5.4.2**. The value of x in Figure 5.5 was taken to be 0.05. The Kimura two Parameter model of evolution was again used, with a transition-transversion ratio of 2. The recombination events occurred three quarters of the way along the exterior branches involved and were of lengths 400 bp, 200 bp and 100 bp. The subset size for the site likelihood calculation was 50 bp.

For each of the three data sets, the posterior probability was found for values of λ ranging between 0 and 1, and for the corresponding MAP estimates of the topology categories. The results are shown in Figure 5.15. In all cases, the posterior probability

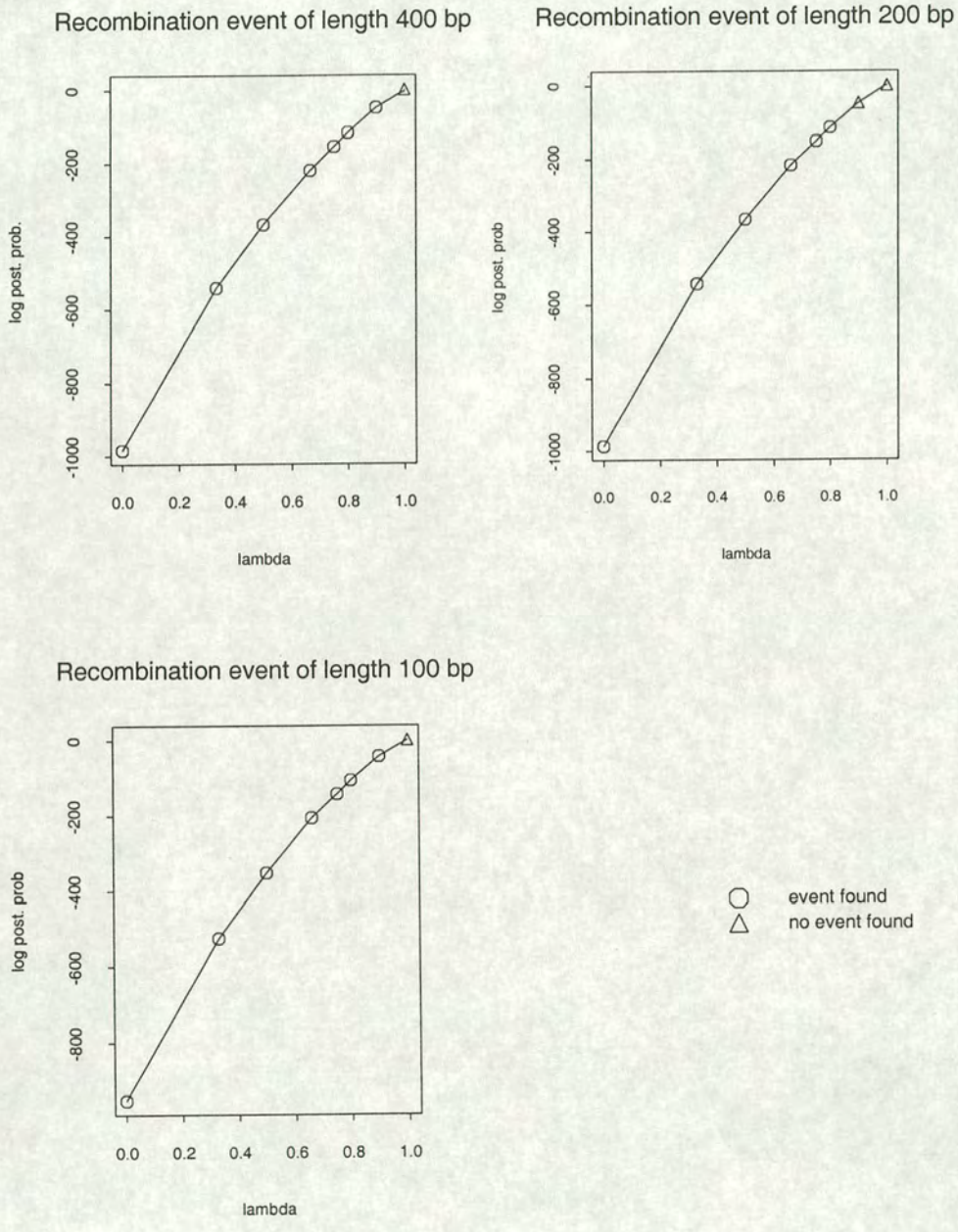


Figure 5.15: The values of the log posterior probability for different values of λ. The triangles mean that the MAP estimate does not find any recombination events.

is highest when $\lambda = 1$. For the highest values of λ, no recombination event is found for any of the data sets, although λ gets very close to one before this happens for the data set containing the 400 bp long recombination event. This arises because the increase in site likelihoods caused by allowing for the recombination event does not offset the very small transition probabilities of change when λ takes on values close to 1. A sufficiently high value of λ will mean that the recombination event is not found by the MAP estimate. Hence, many choices of hyper-prior for λ are likely to lead to a value of $\lambda \approx 1$ being estimated and correspondingly no recombination event would be found. It might be possible to obtain sensible results by using a hyper-prior which places very small probability on λ being high, particularly for data sets with long recombination events, but it is questionable whether this is worth the effort giving the ease of finding the MAP estimate over a range of values of λ, and the insensitivity of the results over a sensible range of λ (i.e., those values which lead to the recombination event being detected). In addition, choosing such a hyper-prior is subjective so that problem is not eliminated.

Finally, a drawback with this procedure is that it only returns a point estimate of a recombination event. Given that it is a Bayesian approach, it would be useful if estimates of credible sets could be found. Monte Carlo Markov Chains appear to be the obvious methodology to use; the problem is coming up with an appropriate procedure. Two approaches, at least, are possible. The first imagines the sequence of topologies as one parameter; a sequence of suitable length is generated from a proposal distribution (e.g., a first-order Markov chain) and is accepted or rejected according to the Metropolis-Hastings algorithm. Initial investigations suggested that this was not a suitable approach, since the chain mixed far too slowly. The other possibility is to consider the topology at each site as a separate parameter and update these in some sequential fashion. This should certainly be investigated.

Chapter 6

Improved Estimation of the Error Bounds for Genetic Distances

In Chapter 2, the modelling of the nucleotide substitution process in DNA sequences using continuous time Markov models was discussed. The derivation of the genetic distance separating two sequences (the average number of changes per position in the sequences) from these Markov models was also described, with the formulae for some of these distance measures given. While many applications merely require a point estimate of the distance [e.g., distance matrix tree reconstruction methods such as Neighbor Joining (Saitou and Nei, 1987) and Least Squares (Cavalli-Sforza and Edwards, 1967; Fitch and Margoliash, 1967; Felsenstein, 1997)], some analyses require an estimate of the variance and/or confidence intervals.

This variance is frequently approximated using the delta method (statistical differentials). If an estimate of a confidence interval is also required, then the distance estimator may be assumed to be normally distributed with a mean equal to the point estimate of the distance and the variance is that yielded by the delta method. However, if the distance estimates are biased and/or skew, then this approximation will not lead to very accurate estimates of the confidence intervals.

This chapter discusses approaches to calculating more accurate approximations to confidence intervals for genetic distances, and finding (continuous) approximations to the sampling distributions. It begins by briefly reviewing the models of nucleotide substitution and the resulting distance estimators which will be used here, and details how the delta method is used to approximate the variance for the distance estimators derived from these models. The two procedures used to approximate the confidence intervals are then discussed: a transformation of normal confidence intervals and the saddlepoint approximation. In the next section, the accuracy of these approximations is examined in a small simulation study. Finally, the methods are applied to some real data sets which have previously been analysed in the literature. Much of the work in this chapter has been previously described in McGuire et al. (1998).

6.1 Models of Nucleotide Substitution

The saddlepoint and transformed normal approximations are illustrated for four different models in this chapter. These models have previously been discussed in **2.5** and include the Felsenstein 84 model (equation 2.9) and its special cases, the Kimura two Parameter model (2.6), the Felsenstein 81 model (2.7) and the Jukes-Cantor model (2.5). Recall that the rate matrix for the Felsenstein 84 (F84) model is given by

$$\mathbf{R}_{F84} = \begin{array}{c|cccc} & A & C & G & T \\ A & - & \gamma\pi_C & \frac{\rho\pi_G}{\pi_A+\pi_G}+\gamma\pi_G & \gamma\pi_T \\ C & \gamma\pi_A & - & \gamma\pi_G & \frac{\rho\pi_T}{\pi_C+\pi_T}+\gamma\pi_T \\ G & \frac{\rho\pi_A}{\pi_A+\pi_G}+\gamma\pi_A & \gamma\pi_C & - & \gamma\pi_T \\ T & \gamma\pi_A & \frac{\rho\pi_C}{\pi_C+\pi_T}+\gamma\pi_C & \gamma\pi_G & - \end{array} . \quad (6.1)$$

The Kimura two Parameter (K2P) model is obtained by setting $\pi_i = 1/4$, $i = A, C, G, T$. If $\rho = 0$, then the rate matrix reduces to that for the Felsenstein 81 (F81) model, while if, in addition, the stationary frequencies are all equal, the rate matrix further reduces to that for the Jukes-Cantor (JC) model.

Note that the other version of a two parameter model, the HKY85 model (equation 2.8) is not suitable for the application of the saddlepoint approximation since a closed form does not exist for the resulting distance estimator (Yang, 1994, see **2.7.1**).

Recall from **2.5**, that the transition-transversion ratio (the sum of all the transition ($A \longleftrightarrow G$, $C \longleftrightarrow T$) rates divided by the sum of all the transversion ($A \longleftrightarrow C$, $A \longleftrightarrow T$, $G \longleftrightarrow C$, $G \longleftrightarrow T$) rates) is an important quantity in two parameter models. For the F84 model, this ratio is given by

$$ts/tv = \frac{\rho A + \gamma B}{\gamma C} \quad (6.2)$$

where

$$\begin{aligned} A &= \frac{\pi_A\pi_G}{\pi_A+\pi_G} + \frac{\pi_C\pi_T}{\pi_C+\pi_T} \\ B &= \pi_A\pi_G + \pi_C\pi_T \\ C &= (\pi_A+\pi_G)(\pi_C+\pi_T). \end{aligned} \quad (6.3)$$

Since the values of A, B and C are known for the K2P model, its transition-transversion ratio has the simpler form

$$ts/tv = \frac{\alpha}{2\beta}. \quad (6.4)$$

The above conditions under which the F84 or K2P rate matrix simplifies to a special case may be expressed in terms of the transition-transversion ratio. If $ts/tv = 0.5$ for the K2P model, then the JC model is obtained, while if $ts/tv = B/C$ for the F84 model, then this model simplifies to the F81 case.

The next section briefly reviews the estimates of genetic distance which may be derived from these models, with an outline of how the F84 distance estimator is obtained. Further details may be found in **2.7.1**.

6.2 Estimators of Genetic Distance

A discussion of genetic distance estimators has previously been given in **2.7.1**. Recall that the additive distance measure most commonly used is the average number of changes that have occurred per site since two sequences diverged. This is equivalent to the product of the overall rate of change (μ) and the time since divergence ($2t$). For the F84 model, the overall rate of change is given by

$$\begin{aligned}\mu &= \pi_A\left[\gamma\pi_C+\gamma\pi_G+\frac{\rho\pi_G}{\pi_A+\pi_G}+\gamma\pi_T\right]+\pi_C\left[\gamma\pi_A+\gamma\pi_G+\gamma\pi_T+\frac{\rho\pi_T}{\pi_C+\pi_T}\right]\\ &\quad+\pi_G\left[\gamma\pi_A+\frac{\rho\pi_A}{\pi_A+\pi_G}+\gamma\pi_C+\gamma\pi_T\right]+\pi_T\left[\gamma\pi_A+\gamma\pi_C+\frac{\rho\pi_C}{\pi_C+\pi_T}+\gamma\pi_G\right].\end{aligned}$$

Following some algebraic manipulation, this becomes

$$\mu = 2A\rho+\gamma(1-\pi_A^2-\pi_C^2-\pi_G^2-\pi_T^2), \tag{6.5}$$

A being defined in (6.3).

In order to find a formula for the F84 distance, the Transition probability (note the use of the capital 'T' to help distinguish between Transition probabilities from a Markov chain and a transition, the biological event) matrix, $\mathbf{P}_{2t}$, of the continuous time Markov chain must first be found. Following the theory presented in **2.5.1**, $\mathbf{P}_{2t} = \exp(2\mathbf{R}t)$, which leads to the following entries in the Transition probability matrix:

$$p_{ij}(2t) = e^{-2(\rho+\gamma)t}\delta_{ij}+e^{-2\gamma t}\left(1-e^{2\rho t}\right)\left(\frac{\pi_j}{\sum_k \pi_k\varepsilon_{jk}}\right)\epsilon_{ij}+\left(1-e^{-2\gamma t}\right)\pi_j \tag{6.6}$$

where

$$\delta_{ij} = \begin{cases}1 & \text{if } i=j\\ 0 & \text{otherwise,}\end{cases}$$

and

$$\varepsilon_{ij} = \begin{cases}1 & \text{if } j \text{ and } k \text{ are either both purines or both pyrimidines}\\ 0 & \text{otherwise.}\end{cases}$$

The expression for the Transition probabilities may be used to find the overall Transition probability that a transition $[P(2t)]$ or a transversion $[Q(2t)]$ occurs in a time interval $2t$. These are specified by

$$P(2t) = 2\left[B+(A-B)e^{-2\gamma t}-Ae^{-2(\rho+\gamma)t}\right], \tag{6.7}$$

$$Q(2t) = 2C\left[1-e^{-2\gamma t}\right]. \tag{6.8}$$

where A, B and C have been previously defined in (6.3).

The F84 distance is given by $2[2A\rho + \gamma(1 - \pi_A^2 - \pi_C^2 - \pi_G^2 - \pi_T^2)]t$. Following some manipulations of equations (6.7) and (6.8), the distance may be expressed as

$$\begin{aligned} d_{F84} = &- 2A \ln\left(1 - \frac{P(2t)}{2A} - \frac{A-B}{2AC}Q(2t)\right) \\ &+ 2(A - B - C)\ln\left(1 - \frac{Q(2t)}{2C}\right). \end{aligned} \tag{6.9}$$

To find the distance between two DNA sequences using the F84 model, the Transition probabilities, $P(2t)$ and $Q(2t)$, may be replaced by their estimates, $\widehat{P}$ and $\widehat{Q}$, from the data set. $\widehat{P}$ and $\widehat{Q}$ represent, respectively, the observed proportion of transitions and transversions between the two sequences. This yields the distance estimate,

$$\widehat{d}_{F84} = -2A\ln\left(1 - \frac{\widehat{P}}{2A} - \frac{A-B}{2AC}\widehat{Q}\right) + 2(A - B - C)\ln\left(1 - \frac{\widehat{Q}}{2C}\right). \tag{6.10}$$

The distance estimator for the K2P model of nucleotide substitution may be derived in a similar manner or may be obtained by substituting the appropriate values of A, B and C, yielding

$$\widehat{d} = -\frac{1}{2}\ln(1 - 2\widehat{P} - \widehat{Q}) - \frac{1}{4}\ln(1 - 2\widehat{Q}). \tag{6.11}$$

For the Jukes-Cantor (JC) and Felsenstein 81 (F81) models, the distance depends on the observed proportion of change, $\widehat{p}$, only since these models have just one rate of change parameter. Thus, the distance estimator has a simpler form given by

$$\widehat{d} = -E\ln(1 - \frac{\widehat{p}}{E}) \tag{6.12}$$

where $E = 1 - \pi_A^2 - \pi_C^2 - \pi_G^2 - \pi_T^2$ (Tajima and Nei, 1982). Note that E is 3/4 for the JC model. A derivation of the distance estimator for the JC model is shown in **2.7.1**.

So far, only point estimates of genetic distances have been given. Some inferences also require an estimate of the error of the estimate. Below a widely-used procedure for estimating the variance of distance estimators is described.

6.3 Estimation of the variance using the delta method

The delta method is a commonly used procedure for estimating variances and was first introduced into the phylogenetic literature by Kimura and Ohta (1972) where they used this method to find an approximation for the variance of the distance estimator from the Jukes-Cantor model. This idea has already been briefly discussed in **2.7.2**, where equation (2.32) gives the general form of a variance estimated using the delta method. As a reminder of this, suppose $\mathbf{V}$ is a statistic with variance-covariance matrix Σ, and let $m(\mathbf{V})$ be a function of $\mathbf{V}$. Then the variance of $m(\mathbf{V})$ may be estimated by

$$\mathrm{Var}[m(\mathbf{V})] \approx \frac{\partial}{\partial \mathbf{v}^T} m(\mathbf{v})\ \Sigma\ \frac{\partial}{\partial \mathbf{v}} m(\mathbf{v})\Big|_{\mathbf{v}=\boldsymbol{\mu}} \tag{6.13}$$

where $\boldsymbol{\mu}$ is the expectation vector of $\mathbf{V}$. For scalar quantities, the variance of $m(V)$ may be simply written as

$$\mathrm{Var}[m(V)] = \sigma^2[m'(v)|_{v=\mu}]^2 \tag{6.14}$$

where $\mu = \mathrm{E}(V)$ and $\sigma^2 = \mathrm{Var}(V)$ and $m'(v)$ denotes the first derivative of $m(v)$ with respect to v.

To find the delta method approximation to the variance for the F81 and JC models, the variance, σ^2, of $\widehat{p}$ must first be found. Now $\widehat{p}$ is an estimator of p, the probability of observing a difference in the nucleotides at a particular site in a two-sequence alignment. If the sequences are n nucleotides long, then the number of differences observed, k, is an observation from a Binomial distribution with parameters n and p. Thus $\widehat{p} = k/n$ and $\mathrm{Var}(\widehat{p}) = p(1-p)/n$ which may be estimated, if necessary, by substituting $\widehat{p}$ for p.

The remaining component to be found in (6.14) is $m'(v)$. Using (6.12) which gives $\widehat{d}$ as a function of $\widehat{p}$, $f(\widehat{p})$, the first derivative with respect to p may be found:

$$f'(\widehat{p}) = \left(1 - \frac{p}{E}\right)^{-1}. \tag{6.15}$$

Substituting the value of the variance of $\widehat{p}$ and the above into (6.14) yields the expression:

$$\mathrm{Var}(\widehat{d}) \approx \frac{p(1-p)}{n(1-p/E)^2} \tag{6.16}$$

where $\widehat{p}$ may be substituted for p if p is unknown (generally the case in practice).

For the F84 and K2P models the observed (bivariate) statistic is $\mathbf{V} = (\widehat{P}, \widehat{Q})^T$. $n\widehat{P}$ and $n\widehat{Q}$ are observations from a multinomial distribution with parameters n, P and Q, where P and Q are the probabilities of observing a transition and a transversion respectively. Equations (6.10) or (6.11) [the former for the F84 model, the latter for the K2P model] express d as a function of $\mathbf{V}$, $f(\mathbf{V})$, so the vector of the first partial derivatives (with respect to P and Q) may be easily found. For the F84 model

$$\begin{aligned}\frac{df(\mathbf{V})}{d\mathbf{V}} = \Bigg(& \left[1 - \tfrac{P}{2A} - \tfrac{(A-B)Q}{2AC}\right]^{-1}, \ \tfrac{A-B}{C}\left[1 - \tfrac{P}{2A} - \tfrac{(A-B)Q}{2AC}\right]^{-1} \\ & - \tfrac{A-B-C}{C}\left[1 - \tfrac{Q}{2C}\right]^{-1}\Bigg)^T\end{aligned} \tag{6.17}$$

while for the K2P model,

$$\frac{df(\mathbf{V})}{d\mathbf{V}} = \left([1-2P-Q]^{-1}, \ \frac{1}{2}[(1-2P-Q)^{-1} + (1-2Q)^{-1}]\right). \tag{6.18}$$

The variance-covariance matrix of $\mathbf{V}$ is simple to find since $\mathbf{V}$ is a statistic from a multinomial distribution. It is simply

$$\Sigma_{\mathbf{V}} = \begin{pmatrix} \frac{P(1-P)}{n} & \frac{-PQ}{n} \\ \frac{-PQ}{n} & \frac{Q(1-Q)}{n} \end{pmatrix}.$$

Substituting these into (6.13) yields

$$\mathrm{Var}(\widehat{d}) \approx \frac{1}{n}[a^2P + b^2Q - (aP + bQ)^2] \tag{6.19}$$

where

$$\begin{aligned} a &= AC/[AC - CP/2 - (A-B)Q/2] \\ b &= A(A-B)/[AC - CP/2 - (A-B)Q/2] - (A-B-C)/(C-Q/2). \end{aligned}$$

If P and Q are unknown, then their estimated values, $\widehat{P}$ and $\widehat{Q}$ may be substituted into (6.19). For the K2P model the variance is approximated by

$$\mathrm{Var}(\widehat{d}) \approx \frac{1}{n}[c^2P + g^2Q - (cP + gQ)^2] \tag{6.20}$$

where c and g, for ease of notation, correspond to the first and second entires respectively in the vector given in (6.18).

Once the variance has been calculated using the delta method, the sampling distribution and confidence intervals may be found by assuming that the genetic distance estimator, $\widehat{d}$, is normally distributed with mean d and variance equal to that found from the delta method. Since the distance is a function of a sum of independent variables (e.g., whether a difference is observed at a particular position or not for an F81 distance), the sampling distribution of the distance estimator should approach a normal distribution as the sequence length increases by the Central Limit theorem. For shorter sequence lengths, however, the assumption of normality may be questionable. This approximation is later referred to as the *normal-delta* approximation.

6.3.1 Other approaches to the estimation of confidence intervals

Other more complicated (either computationally or mathematically) approaches may be used to calculate confidence intervals for genetic distances (see **2.7.2**). These include using the bootstrap to yield an approximation to the sampling distribution of the distance estimator which has the disadvantage of a high computational burden.

Andrieu et al. (1997) suggested using interval estimation to calculate the exact confidence intervals for the JC and K2P models; this may also be used for the F84 and F81 models. Details of the procedure have been given in **2.7.2**. This method is particularly useful where no change, or very little has occurred as sampling theory is unhelpful in this case. However, it is a somewhat computationally tedious approach; for reasonable amounts of change it may be possible to find approximations which perform well, and yet are easier to calculate. In addition, for two parameter models, it requires that the transition-transversion ratio be assumed to be known, something which is very unlikely to be the case. Furthermore, for computational reasons it is difficult to extend it to three parameter models; this would also require restrictive assumptions about

the parameters, similar to the assumption of the transition-transversion ratio for a two parameter model.

The following parts of this chapter propose two approximations. The first may be used only in a few limited cases, while the second has a wider range of applicability.

6.4 A very accurate approximation to the true confidence intervals of the F81 and JC distance estimators

The distance estimator, $\widehat{d}$, for the F81 and JC models (equation 6.12) is a simple function of $\widehat{p} = k/n$, where k is the number of differences observed between the two nucleotide sequences, and n is the sequence length. Clearly, k is an observation from a binomial distribution, $B(n,p)$, where p is the true probability of observing a difference. Hence, the sampling distribution of $\widehat{p}$ is well approximated by a normal distribution with mean p and variance $p(1-p)/n$, provided $\min\{np, n(1-p)\}$ is not small (typically the smaller of the two should be greater than 5; Clarke and Cooke, 1992, p. 237). Therefore, finding confidence intervals for $\widehat{p}$ is a straightforward task.

Since $\widehat{d}$ is a monotone function of $\widehat{p}$ (equation 6.12), it is possible to transform confidence intervals for $\widehat{p}$ to obtain the corresponding intervals for $\widehat{d}$. If the lower bound of the $100(1-\alpha)\%$ confidence interval for $\widehat{p}$ is $lb^p_{\alpha/2}$, and the upper bound is $ub^p_{\alpha/2}$ then the corresponding lower ($lb^d_{\alpha/2}$) and upper ($ub^d_{\alpha/2}$) bounds of the $100(1-\alpha)\%$ confidence interval for $\widehat{d}$ are given by

$$lb^d_{\alpha/2} = -E\ln(1 - lb^p_{\alpha/2}/E)$$

and

$$ub^d_{\alpha/2} = -E\ln(1 - ub^p_{\alpha/2}/E). \tag{6.21}$$

This approximation is later referred to as the *transformed normal* approximation.

It is also possible to use this method to approximate the sampling distribution of $\widehat{d}_{F81}$. Strictly speaking, this will not be correct since $\widehat{d}_{F81}$, for a given data set, has a discrete distribution, while transforming a normal distribution will lead to a continuous sampling distribution. Nevertheless, such an approximation is useful to examine the shape of the sampling distribution (i.e., its bias, skewness etc.).

As mentioned above, in most cases the sampling distribution of $\widehat{p}$ is well approximated by a normal distribution, having the form

$$f(\widehat{p}) = (2\pi\sigma^2)^{-1/2}\exp\left\{-\frac{(\widehat{p}-p)^2}{2\sigma^2}\right\} \tag{6.22}$$

where $\sigma^2 = p(1-p)/n$. To find the sampling distribution of $\widehat{d}$, the sampling distribution of $\widehat{p}$ may be transformed as follows:

$$g(\widehat{d}) = f[h^{-1}(\widehat{d})]\frac{d\widehat{p}}{d\widehat{d}}. \tag{6.23}$$

From (6.12),

$$\begin{aligned}\widehat{p} &= h^{-1}(\widehat{d}) \\ &= E(1 - e^{-\widehat{d}/E})\end{aligned} \tag{6.24}$$

and therefore

$$\frac{d\widehat{p}}{d\widehat{d}} = e^{-\widehat{d}/E}. \tag{6.25}$$

Substituting (6.24) and (6.25) into (6.23) yields the following approximation to the sampling density of $\widehat{d}_{F81}$, $g(\widehat{d})$:

$$g(\widehat{d}) \approx \widetilde{g}(\widehat{d}) = (2\pi\sigma^2)^{-1/2} \exp\left\{-[E(1 - e^{-\widehat{d}/E}) - p]^2/2\sigma^2\right\} e^{-\widehat{d}/E}. \tag{6.26}$$

This approximation (the transformed normal approximation) to the sampling distribution may be easily plotted using packages such as S-plus (version 3.4, StatSci Inc., Seattle, Washington) or MAPLE (MAPLE V release 4, Waterloo Maple Software, Waterloo).

As the JC model is simply a special case of the F81 model (with E taking the value 3/4 in equation 6.12), this also provides an almost exact approximation to the sampling distribution of the distance estimator for the JC model. The sampling distribution is given by (6.26) and the confidence interval bounds may be found as described above.

This procedure of transforming one distribution to obtain the distribution of another cannot be applied to the other models discussed (K2P and F84), since these consider transitional and transversional changes separately. Thus the underlying distribution of observed changes is multivariate (multinomial) and cannot be transformed to give the sampling distribution of the scalar quantity, $\widehat{d}$. Consequently, confidence intervals for these models cannot be found using this method either. In this case the saddlepoint approximation is the suggested method. Below, some of the historical development and theory behind this approximation are outlined, followed by details of its application to genetic distance estimators.

6.5 Saddlepoint Theory

Some of the background theory of the saddlepoint approximation is described here, beginning with Daniels' (1954) work on an approximation to the mean of n independent, identically distributed random variables followed by generalisations of this technique introduced by Easton and Ronchetti (1986) and Gatto and Ronchetti (1996).

6.5.1 Mean of n independent, identically distributed random variables

Daniels (1954) introduced the saddlepoint technique into the statistics literature by deriving a very accurate approximation to the mean of n independent, identically distributed (i.i.d.) random variables. An outline of his derivation is as follows. Let $X_1, X_2, \ldots, X_n$ be continuous i.i.d. random variables, with cumulative distribution function $F(x)$ and density $f(x)$. Then the moment generating function (mgf) is defined as

$$M_X(\theta) = e^{K(\theta)} = \int_{-\infty}^{\infty} e^{\theta x} f(x)\, dx$$

and suppose the mgf converges for real θ in some non-vanishing interval containing the origin. Let $-c_1 < \theta < c_2$ be the largest such interval ($0 \leq c_1 \leq \infty$, $0 \leq c_2 \leq \infty$, $c_1 + c_2 > 0$).

Consider the mgf of $\bar{X}$ at $i\theta$ or alternatively, the characteristic function (cf) at θ. The cf is given by

$$\begin{aligned} \mathrm{E}[e^{i\theta \sum X_i/n}] &= \prod_{i=1}^{n} \mathrm{E}[e^{i\theta X_i/n}] \\ &= \mathrm{E}[e^{i\theta X/n}]^n \\ &= \mathrm{M}_X(i\theta/n)^n \end{aligned}$$

by the i.i.d. properties. This may be rewritten as $\mathrm{M}_X(it)^n$. Hence, the sampling distribution of $\bar{X}$, $f_n(\bar{x})$ may be found from the inverse Fourier transform

$$\begin{aligned} f_n(\bar{x}) &= \frac{n}{2\pi} \int_{-\infty}^{\infty} M^n(it) e^{-nit\bar{x}}\, dt \\ &= \frac{n}{2\pi} \int_{-\infty}^{\infty} e^{n[K(it) - it\bar{x}]}\, dt. \end{aligned}$$

Equivalently, through a change in variable ($T = it$, $dT = i\, dt$)

$$f_n(\bar{x}) = \frac{n}{2\pi i} \int_{-i\infty}^{i\infty} e^{n[K(T) - T\bar{x}]}\, dT.$$

This integral is the same as

$$f_n(\bar{x}) = \frac{n}{2\pi i} \int_{\tau - i\infty}^{\tau + i\infty} e^{n[K(T) - T\bar{x}]}\, dT \tag{6.27}$$

where τ is some real number within the strip of convergence of $\mathrm{M}(T)$.

When n is large, $f_n(\bar{x})$ may be approximated by choosing the path of integration to pass through a saddlepoint of the integrand in such a way that the integrand is negligible outside the immediate neighbourhood. The saddlepoints are situated where the exponent has zero derivative, i.e., where

$$K'(T) = \bar{x}, \tag{6.28}$$

$K'(T)$ being the first derivative of $K(T)$ with respect to T. Under general conditions, (6.28) has a single real root, T_0, within the strip of convergence of $M_X(t)$, ($-c_1$, c_2) for every value of $\bar{x}$ such that $0 < F_n(\bar{x}) < 1$, where $F_n(\bar{x})$ is the cumulative distribution function of $\bar{X}$.

Since T_0 is a minimum of $K(T) - T\bar{x}$ for real T, the modulus of the integrand must have a maximum at T_0 on the chosen path of integration. It can be shown (Daniels, 1954) that on any admissible straight line parallel to the imaginary axis, the integrand attains its maximum modulus only where the line crosses the real axis (essentially, it is shown that for the line $T = \tau + iy$ (τ real), $|M(T)e^{-T\bar{x}}| < |M(\tau)e^{-\tau\bar{x}}|$). By the Riemann-Lebesgue Lemma, $M(\tau + iy) = O(|y|^{-1})$, so the integrand cannot approach arbitrarily near its maximum as $|y|$ becomes large. So for the particular path of integration chosen, only the neighbourhood of T_0 need be considered.

On the contour near T_0, the Taylor expansion of $K(T) - T\bar{x}$ at T_0 is

$$\begin{aligned} K(T) - T\bar{x} &= K(T_0) - T_0\bar{x} + (T - T_0)(K'(T_0) - \bar{x}) + \frac{1}{2}(T - T_0)^2 K''(T_0) \\ &\quad + \frac{1}{3!}(T - T_0)^3 K'''(T_0) + \frac{1}{4!}(T - T_0)^4 K^{iv}(T_0) + \ldots \end{aligned} \tag{6.29}$$

Since $T = T_0 + iy$, $T - T_0 = iy$, (6.29) becomes

$$\begin{aligned} K(T) - T\bar{x} = K(T_0) - T_0\bar{x} - \frac{1}{2}y^2 K''(T_0) - \\ \frac{1}{6}iy^3 K'''(T_0) + \frac{1}{24}y^4 K^{iv}(T_0) + \ldots \end{aligned} \tag{6.30}$$

Making a change of variable ($y = v/[nK''(T_0)]^{1/2}$ so $dy = [nK''(T_0)]^{-1/2}dv$) in (6.30) and putting this into (6.27) yields

$$\begin{aligned} f_n(\bar{x}) &= \frac{n}{2\pi}\frac{1}{\sqrt{nK''(T_0)}}\int_{-\infty}^{\infty} \exp\left\{ n\left[K(T_0) - T_0\bar{x} - \frac{1}{2}K''(T_0)\frac{v^2}{nK''(T_0)} \right.\right. \\ &\quad \left.\left. -\frac{1}{6}K'''(T_0)\frac{iv^3}{[nK''(T_0)]^{3/2}} + \frac{1}{24}K^{iv}(T_0)\frac{v^4}{n^2K''(T_0)^2} + \ldots \right]\right\} dv. \end{aligned}$$

Letting $\lambda_j(T) = K^{(j)}(T)/[K''(T)]^{j/2}$, the exponent becomes

$$\exp\{n[K(T_0) - T_0\bar{x}]\} \exp\left\{ \frac{-1}{2}v^2 - \frac{1}{6}iv^3\frac{\lambda_3(T_0)}{\sqrt{n}} + \frac{1}{24}v^4\frac{\lambda_4(T_0)}{n} + \ldots \right\}. \tag{6.31}$$

The second exponential term in this expression has the form $e^{a+\delta}$ where $a = -v^2/2$ and δ consists of the remaining terms which are small. Hence, the Taylor series expansion

$$\begin{aligned} e^{a+\delta} &= e^a + \delta e^a + \frac{\delta^2}{2}e^a + \ldots \\ &= e^a\left(1 + \delta + \frac{\delta^2}{2} + \ldots\right) \end{aligned}$$

may be applied to the second exponential in (6.31), yielding

$$\begin{aligned} &e^{-v^2/2}\left(1 - \frac{1}{6}iv^3\frac{\lambda_3(T_0)}{\sqrt{n}} + \frac{1}{24}v^4\frac{\lambda_4(T_0)}{n} + \frac{1}{2}(-\frac{1}{36}v^6\frac{\lambda_3^2(T_0)}{n} + \ldots\right) \\ &\quad = e^{-v^2/2}\left(1 - \frac{1}{6}\lambda_3(T_0)\frac{iv^3}{\sqrt{n}} + \frac{1}{n}\left[\frac{1}{24}\lambda_4(T_0)v^4 - \frac{1}{72}\lambda_3^2(T_0)v^6 + \ldots\right]\right). \end{aligned}$$

Therefore,

$$f_n(\bar{x}) \approx \frac{1}{2\pi}\left[\frac{n}{K''(T_0)}\right]^{1/2} e^{n[K(T_0)-T_0\bar{x}]} \times \int_{-\infty}^{\infty} e^{-v^2/2}\left(1 - \frac{1}{6}iv^3\frac{\lambda_3(T_0)}{\sqrt{n}} + \frac{1}{n}\left[\frac{1}{24}\lambda_4(T_0)v^4 - \frac{1}{72}\lambda_3^2(T_0)v^6\right]\right) dv. \quad (6.32)$$

The odd powers of v in (6.32) are oscillating (odd) complex functions, so their integrals are zero. The even functions may be integrated by parts:

$$\int_{-\infty}^{\infty} e^{-v^2/2}\, dv = \sqrt{2\pi}$$

and

$$\begin{aligned}\int_{-\infty}^{\infty} 1.e^{-v^2/2}\, dv &= [e^{-v^2/2}v]_{-\infty}^{\infty} - \int_{-\infty}^{\infty} e^{-v^2/2}(-v)v\, dv \\ &= \int_{-\infty}^{\infty} v^2 e^{-v^2/2}\, dv.\end{aligned}$$

So

$$\int_{-\infty}^{\infty} v^2 e^{-v^2/2}\, dv = \sqrt{2\pi}$$

and in a similar manner

$$\begin{aligned}\int_{-\infty}^{\infty} v^4 e^{-v^2/2}\, dv &= 3\sqrt{2\pi} \\ \int_{-\infty}^{\infty} v^6 e^{-v^2/2}\, dv &= 15\sqrt{2\pi}.\end{aligned}$$

Therefore, upon integration, (6.32) becomes

$$\begin{aligned} f_n(\bar{x}) &\approx \frac{1}{2\pi}\left[\frac{n}{K''(T_0)}\right]^{1/2} e^{n[K(T_0)-T_0\bar{x}]} \\ &\quad \times \left[\sqrt{2\pi} + \frac{1}{n}\left(\frac{1}{24}\lambda_4(T_0)3\sqrt{2\pi} - \frac{1}{72}\lambda_3^2(T_0)15\sqrt{2\pi}\right) + \ldots\right] \\ &= \left[\frac{n}{2\pi K''(T_0)}\right]^{1/2} e^{n[K(T_0)-T_0\bar{x}]}\left\{1 + \frac{1}{n}\left[\frac{1}{8}\lambda_4(T_0) - \frac{5}{24}\lambda_3^2(T_0)\right] + \ldots\right\}. \end{aligned}$$

Thus

$$g_n(\bar{x}) = \left[\frac{n}{2\pi K''(T_0)}\right]^{1/2} e^{n[K(T_0)-T_0\bar{x}]} \quad (6.33)$$

is the saddlepoint approximation to $f_n(\bar{x})$, with error of order n.

6.5.2 Saddlepoint approximations to general statistics

Easton and Ronchetti (1986) generalised this procedure to deal with general statistics. As above, let $X_1, X_2, \ldots, X_n$ be i.i.d. random variables with density $f(x)$, and

let $V_n(X_1, X_2, \ldots, X_n)$ be a real-valued statistic with density, $f_n(x)$. Let $M_n(t) = \int e^{tx} f_n(x)\, dx$ be the moment generating function of $f_n(x)$, and $K_n(t) = \log M_n(t)$ be the cumulant generating function. Then $f_n(x)$ can be expressed in terms of the Fourier inversion formula:

$$\begin{aligned} f_n(x) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} M_n(it) e^{-itx}\, dt \\ &= \frac{i}{2\pi i} \int_{-i\infty}^{i\infty} M_n(nT) e^{-nTx}\, dT \\ &= \frac{n}{2\pi i} \int_{\tau-i\infty}^{\tau+i\infty} e^{n[R_n(T)-Tx]}\, dT \end{aligned} \tag{6.34}$$

where τ is any real number in the interval of convergence of the moment generating function, and

$$R_n(T) = K_n(nT)/n. \tag{6.35}$$

Note that if V_n is the arithmetic mean, then $R_n(T) = K(T)$, the cumulant generating function of $f(x)$, and thus (6.34) is the same as (6.27). In the general case, $R_n(T)$ must be approximated, and then the saddlepoint method of asymptotic analysis may be applied to (6.34), following similar steps to those described above and detailed in Daniels (1954).

If an Edgeworth expansion, $\widetilde{f}_n(x)$, for $f_n(x)$ up to, and including the term of order n^{-1} is available, then it is possible to obtain an approximation, $\widetilde{R}_n(T)$, for $R_n(T)$ in terms of the first four cumulants:

$$\widetilde{R}_n(T) = \mu_n T + \frac{n\sigma_n^2 T^2}{2} + \frac{n^2 \kappa_{3n} \sigma_n^3 T^3}{6} + \frac{n^3 \kappa_{4n} \sigma_n^4 T^4}{24} \tag{6.36}$$

where μ_n is the mean, σ_n^2 is the variance, and κ_{3n} and κ_{4n} are the third and fourth order standardised cumulants respectively of V_n. Applying the saddlepoint technique, as described above, yields the saddlepoint approximation to the density at a value x:

$$f_n(x) \approx \left[\frac{n}{2\pi \widetilde{R}''(T_0)}\right]^{1/2} e^{n[\widetilde{R}(T_0)-T_0 x]} \tag{6.37}$$

which has uniform error of order n^{-1}. As before, T_0 is the saddlepoint, found by solving the equation:

$$\widetilde{R}_n'(T_0) = x. \tag{6.38}$$

Since $\widetilde{R}_n'(T)$ is a third-degree polynomial, the existence of a unique saddlepoint, T_0, may easily be shown.

6.5.3 Marginal Densities and Tail Area Probabilities

Gatto and Ronchetti (1996) derived the saddlepoint approximations of marginal densities and tail area probabilities of general non-linear statistics, based on the expansion of the statistic up to the second order. It is this technique which may be applied to the problem of inference for the genetic distances considered in this chapter.

Once again, consider n i.i.d. random variables, $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n$, possibly multivariate, with cumulative distribution function (cdf), F, and a (possibly multivariate) statistic $\mathbf{T}_n(\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n)$. Let $\mathbf{v}_0 = \mathbf{V}(F)$ be the statistical functional defined by $\mathbf{V}_n = \mathbf{V}(F^{(n)})$, where $F^{(n)}$ is the empirical cdf. Suppose it is of interest to make inferences about a real-valued function, $m(\mathbf{V}_n)$, with continuous and nonzero gradient at $\mathbf{v}_0$, and continuous second derivative at $\mathbf{v}_0$.

A Taylor series expansion is used to approximate $m(\mathbf{V}_n) - m(\mathbf{v}_0)$:

$$\begin{aligned} m(\mathbf{V}_n) - m(\mathbf{v}_0) &= (\mathbf{V}_n - \mathbf{v}_0)^T \frac{\partial}{\partial \mathbf{v}} m(\mathbf{v})\Big|_{\mathbf{v}=\mathbf{v}_0} \\ &\quad + \frac{1}{2}(\mathbf{V}_n - \mathbf{v}_0)^T \frac{\partial^2}{\partial \mathbf{v} \partial \mathbf{v}^T} m(\mathbf{v})\Big|_{\mathbf{v}=\mathbf{v}_0} (\mathbf{V}_n - \mathbf{v}_0) \\ &\quad + O_p(n^{-3/2}). \end{aligned} \tag{6.39}$$

The Von Mises expansion of the statistic, $\mathbf{V}_n$, up to the second-order term is found:

$$\begin{aligned} \mathbf{V}_n - \mathbf{v}_0 = \frac{1}{n} \sum_{i=1}^{n} \mathbf{k}_1(\mathbf{X}_i; F) \\ + \frac{1}{2n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \mathbf{k}_2(\mathbf{X}_i, \mathbf{X}_j; F) + O_p(n^{-3/2}). \end{aligned} \tag{6.40}$$

This may be substituted into (6.39), leading to the following quadratic approximation, U_n, to $m(\mathbf{V}_n) - m(\mathbf{v}_0)$:

$$U_n = \frac{2}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} h(\mathbf{X}_i, \mathbf{X}_j)$$

where

$$\begin{aligned} h(\mathbf{x}_i, \mathbf{x}_j) = \Bigg\{ \left[\mathbf{k}_1^T(\mathbf{x}_i; F) + \mathbf{k}_1^T(\mathbf{x}_j; F) + \mathbf{k}_2^T(\mathbf{x}_i, \mathbf{x}_j; F) \right] \frac{\partial m(\mathbf{v})}{\partial \mathbf{v}}\Bigg|_{\mathbf{v}=\mathbf{v}_0} \\ + \mathbf{k}_1^T(\mathbf{x}_i; F) \frac{\partial^2 m(\mathbf{v})}{\partial \mathbf{v} \partial \mathbf{v}^T}\Bigg|_{\mathbf{v}=\mathbf{v}_0} \mathbf{k}_1^T(\mathbf{x}_j; F) \Bigg\} \Big/ 2 \end{aligned}$$

and U_n is a U statistic of degree 2. This statistic may be expanded by means of an Edgeworth expansion, and thus estimates of the cumulants may be obtained. This leads to an estimate of $R_n(T)$ so that (6.37) may be used to calculate the saddlepoint approximation.

There are two steps in the procedure to estimate the cumulants used to calculate $\widetilde{R}_n(T)$:

Step 1 The quantities $g(\mathbf{X})$, $\gamma(\mathbf{X}_1, \mathbf{X}_2)$ and σ_g^2 are calculated from the first and second order kernels, $\mathbf{k}_1$ and $\mathbf{k}_2$ respectively, of the Von Mises expansion of $\mathbf{V}_n$ (see equation 6.40):

$$g(\mathbf{x}) = \frac{1}{2}\mathbf{k}_1^T(\mathbf{x}; F)\frac{\partial}{\partial \mathbf{v}}m(\mathbf{v})\Big|_{\mathbf{v}=\mathbf{v}_0} \tag{6.41}$$

$$\gamma(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{2}\Big[\mathbf{k}_2^T(\mathbf{x}_1, \mathbf{x}_2; F)\frac{\partial}{\partial \mathbf{v}}m(\mathbf{v})\Big|_{\mathbf{v}=\mathbf{v}_0} + \mathbf{k}_1^T(\mathbf{x}_1; F)\frac{\partial^2}{\partial \mathbf{v}\partial \mathbf{v}^T}m(\mathbf{v})\Big|_{\mathbf{v}=\mathbf{v}_0}\mathbf{k}_1(\mathbf{x}_2; F)\Big] \tag{6.42}$$

$$\sigma_g^2 = \mathrm{E}[g^2(\mathbf{X}_1)]. \tag{6.43}$$

Step 2 Using the quantities above in equations (6.41) to (6.43), approximations to the mean, μ_n (this will often be zero), variance, σ_n^2, and the standardised cumulants, κ_{3n} and κ_{4n}, of $m(\mathbf{V}_n)$ may then be computed using

$$\begin{aligned}
\mu_n &= \frac{1}{n}\mathrm{E}[\gamma(\mathbf{X}_1, \mathbf{X}_2)] \\
\sigma_n^2 &= 4\sigma_g^2/n + 2\mathrm{E}[\gamma^2(\mathbf{X}_1, \mathbf{X}_2)]/[n(n-1)] \\
\kappa_{3n} &= n^{-1/2}\sigma_g^{-3}\{\mathrm{E}[g^3(\mathbf{X}_1)] + 3\mathrm{E}[g(\mathbf{X}_1)g(\mathbf{X}_2)\gamma(\mathbf{X}_1, \mathbf{X}_2)]\} \\
\kappa_{4n} &= n^{-1}\sigma_g^{-4}\{\mathrm{E}[g^4(\mathbf{X}_1)] - 3\sigma_g^4 + 12\mathrm{E}[g^2(\mathbf{X}_1)g(\mathbf{X}_2)\gamma(\mathbf{X}_1, \mathbf{X}_2)] \\
&\quad + 12\mathrm{E}[g(\mathbf{X}_1)g(\mathbf{X}_2)\gamma(\mathbf{X}_1, \mathbf{X}_3)\gamma(\mathbf{X}_2, \mathbf{X}_3)]\}
\end{aligned} \tag{6.44}$$

where all the expectations are taken with respect to F.

These approximations may be used to find $\widetilde{R}_n(T)$ in (6.36), and consequently the saddlepoint approximation to the density, (6.37) may be calculated.

Gatto and Ronchetti (1996) also give expressions for the tail area probability:

$$\begin{aligned}
1 - G_n(x) &= P(m(\mathbf{V}_n) - m(\mathbf{v}_0) > x) \\
&= 1 - \Phi(r) + \phi(r)\left[\frac{1}{s} - \frac{1}{r}\right]
\end{aligned} \tag{6.45}$$

where

$$s = T_0[n\widetilde{R}_n''(T_0)]^{1/2} \tag{6.46}$$

$$r = sgn(T_0)\{2n[T_0 x - \widetilde{R}_n(T_0)]\}^{1/2}, \tag{6.47}$$

$\phi(\cdot)$ and $\Phi(\cdot)$ are the density and distribution functions of the standard normal respectively, and T_0 is the saddlepoint given by the solution of (6.38).

The approximation to the tail area probability (6.45) is for continuous variables. Daniels (1987) considers the problem for lattice variables and notes that the form of the tail area probability is the same as that for continuous variables. The difference is in the definition of s. For lattice variables, this has the form

$$s = \left(1 - e^{-T_0}\right)\left[n\widetilde{R}_n''(T_0)\right]^{1/2}. \tag{6.48}$$

Note that r remains as defined in (6.47). The saddlepoint is still given by the solution of (6.38).

It is also possible to incorporate a continuity correction into the formula for the tail probability of a lattice variable. Again, only the definition of s changes, becoming

$$s = 2\sinh\left(\frac{T_0}{2}\right)\left[n\widetilde{R}_n''(T_0)\right]^{1/2}. \tag{6.49}$$

The definitions of s for a lattice variable (6.48) and for a lattice variable incorporating a continuity correction (6.49) will need to be considered when deriving the saddlepoint approximation for the F81 distance estimator. While incorporating the continuity correction may appear to be the sensible choice, Daniels (1987) notes that the uncorrected form performed better for the Poisson distribution.

6.6 Application of the saddlepoint approximation to the tail probabilities of distance estimators

As has been indicated above, the technique for marginal densities for general non-linear statistics developed by Gatto and Ronchetti (1996) is used here to more accurately estimate the tail probabilities and the sampling distribution of a distance estimator. Since the four models may be put into two classes: the one parameter models (F81 and JC) and the two parameter ones (F84 and K2P), two sets of computations must be done. The simpler task of finding the saddlepoint approximation for the F81 and JC models is shown first, followed by the computations necessary to find the approximation for the two parameter models.

6.6.1 Saddlepoint approximations for the JC and F81 distance estimators

Recall from (6.12) in **6.2** that the estimator of genetic distance for the JC and F81 models is

$$\widehat{d} = -E\ln\left(1 - \frac{\widehat{p}}{E}\right)$$

where

$$E = 1 - \pi_A^2 - \pi_C^2 - \pi_G^2 - \pi_T^2.$$

For the JC model E takes the value 0.75. Putting this in the framework of the saddlepoint approximation, the (scalar) random variables, X_i, correspond to observations from a Bernoulli distribution with parameter p, where p is the probability that a nucleotide substitution is observed, while the statistic V_n becomes the proportion of observed changes ($\widehat{p}$) between the two sequences. Hence, $v_0 = p$. In addition, $m(V_n)$ is

the function $\widehat{d} = f(\widehat{p})$. Now the steps outlined above may be used to calculate the saddlepoint approximation with the kernels k_1 and k_2 being specified by

$$k_1(x;F) = \begin{cases} 1-p & \text{change observed} \quad (\text{prob} = p) \\ -p & \text{no change} \qquad\quad (\text{prob} = 1-p) \end{cases}$$

while $k_2(x_i, x_j; F)$ is zero.

The derivatives in (6.41) and (6.42) are quite simple, being

$$\frac{d}{dp} f(p) = \left(1 - \frac{p}{E}\right)^{-1}$$

and

$$\frac{d^2}{dp^2} f(p) = \left(E\left[1 - \frac{p}{E}\right]^2\right)^{-1}.$$

These may be substituted into (6.41) and (6.42). Once these expressions have been found, the rest of the calculations used to estimate $\widetilde{R}_n(T)$ may be carried out in a straightforward manner.

An approximation to the sampling distribution may be found using (6.37). However, as noted in **6.4**, this is not strictly correct since the true sampling distribution is discrete. The approximation to the tail area probabilities is found using (6.45). Since $\widehat{p}$ is a lattice variable, the tail probability for lattice variables must be used in either the uncorrected (6.48) or continuity corrected forms (6.49). Initial investigations suggested that the uncorrected form gave more accurate estimates for shorter sequence lengths (for longer sequences, both forms converge to each other). Thus, the uncorrected form has been used here.

6.6.2 Saddlepoint approximations to the tail probabilities of the K2P and F84 distance estimators

Recall that the expression for the genetic distance estimator for the F84 model is given by

$$\widehat{d} = -2A \ln\left(1 - \frac{\widehat{P}}{2A} - \frac{(A-B)\widehat{Q}}{2AC}\right) + 2(A-B-C)\ln\left(1 - \frac{\widehat{Q}}{2C}\right)$$

where

$$\begin{aligned} A &= \frac{\pi_A\pi_G}{\pi_A+\pi_G} + \frac{\pi_C\pi_T}{\pi_C+\pi_T} \\ B &= \pi_A\pi_G + \pi_C\pi_T \\ C &= (\pi_A+\pi_G)(\pi_C+\pi_T) \end{aligned}$$

(see **6.1**). For the K2P model, A and C are 0.25 while B is 0.125.

For the F84 and K2P models, the random variables, $\mathbf{X}_i$, are bivariate, taking on three possible values: $(0\ ,\ 0)^T$ if no change has occurred; $(1\ ,\ 0)^T$ if a transitional change

has occurred; $(0\ ,\ 1)^T$ if a transversional change has occurred. Thus the underlying distribution, F, is multinomial, $M(1, P, Q)$, where P is the probability of observing a transition, and Q is the probability of observing a transversion. $\mathbf{V}_n(F^{(n)})$ corresponds to the bivariate statistic $(\widehat{P}\ ,\ \widehat{Q})^T$, where $\widehat{P}$ and $\widehat{Q}$ represent the observed proportion of transitions and transversions respectively. Consequently, $\mathbf{v}_0 = (P\ ,\ Q)^T$. The saddlepoint approximation may be calculated as described above, and as before, $\mathbf{k}_2(\cdot)$ is zero while

$$\mathbf{k}_1(\mathbf{x}; F) = \begin{cases} (1-P\ ,\ -Q)^T & \text{transition} \quad (\text{prob} = P) \\ (-P\ ,\ 1-Q)^T & \text{transversion} \quad (\text{prob} = Q) \\ (-P\ ,\ -Q)^T & \text{no change} \quad (\text{prob} = 1-P-Q) \end{cases}$$

For these models, the partial derivatives in (6.41) and (6.42) are more complicated. For the F84 model

$$\frac{\partial}{\partial \mathbf{v}} m(\mathbf{v}) = \left(\left[1 - \frac{P}{2A} - \frac{(A-B)Q}{2AC} \right]^{-1}, \right.$$
$$\left. \frac{A-B}{C} \left[1 - \frac{P}{2A} - \frac{(A-B)Q}{2AC} \right]^{-1} - \frac{A-B-C}{C} \left[1 - \frac{Q}{2C} \right]^{-1} \right)^T.$$

This simplifies to

$$\frac{\partial}{\partial \mathbf{v}} m(\mathbf{v}) = \left([1-2P-Q]^{-1}\ ,\ \frac{1}{2}\left[(1-2P-Q)^{-1} + (1-2Q)^{-1} \right] \right)$$

for the K2P model. The matrix of second order derivatives for both models has the following form:

$$\frac{\partial^2}{\partial \mathbf{v} \partial \mathbf{v}^T} m(\mathbf{v}) = \begin{pmatrix} a & b \\ b & c \end{pmatrix} \tag{6.50}$$

where

$$a = \frac{1}{2A} \left[1 - \frac{P}{2A} - \frac{(A-B)Q}{2AC} \right]^{-2}$$
$$b = \frac{A-B}{2AC} \left[1 - \frac{P}{2A} - \frac{(A-B)Q}{2AC} \right]^{-2}$$
$$c = \frac{(A-B)^2}{2AC^2} \left[1 - \frac{P}{2A} - \frac{(A-B)Q}{2AC} \right]^{-2} - \frac{A-B-C}{2C^2} \left[1 - \frac{Q}{2C} \right]^{-2}$$

for the F84 model, and

$$a = 2(1-2P-Q)^{-2}$$
$$b = (1-2P-Q)^{-2}$$
$$c = \frac{1}{2}(1-2P-Q)^{-2} + (1-2Q)^{-2}$$

for the K2P model. Once these derivatives have been found, the saddlepoint calculations are again straightforward, although more complicated than those for the one parameter models.

Programs to carry out these computations have been written in C. Currently these programs run interactively, requiring the user to specify the range of values of the saddlepoint, T_0 and the number of points within this range over which to evaluate the tail probabilities. The program outputs the location corresponding to each T_0 (x in equation 6.38) and the tail probability at that point (using equation 6.45). The lattice version is used for the F81 and JC models, while the continuous version of (6.45) is appropriate for the F84 and K2P models. Since the calculations appear to require negligible computer time, confidence intervals may be found quickly by sensible choices of the range and number of points over which to evaluate the tail probability.

6.7 Evaluation of Saddlepoint approximation

Two approximations to the tail probabilities of some genetic distance estimators have been developed above. The first, based on transforming the normal approximation to the binomial, should be very accurate in most practical applications as the normal approximation to the binomial can be good even when $\min\{np, n(1-p)\}$ is as small as 5. The lengths of DNA sequences used in practice are such that this condition is usually satisfied easily. Therefore, it is expected that any investigation into its performance should return positive results.

It is difficult to make a similar claim about the saddlepoint approximation. While this is, in general, a very accurate approximation, it cannot be guaranteed that it will perform well in every situation. Therefore, it is necessary to study the performance of this approximation under a variety of conditions to see if it is a significant improvement on the existing normal-delta approximation (see **6.3**). Thus, a small simulation study was carried out, the intervals from the saddlepoint and normal-delta approximations being compared to the exact answers. The details of this simulation study are given below.

6.7.1 Details and Results of the Simulation Study

The simulation study may be broken down into two sections. For the purposes of this chapter, the 95% confidence intervals are calculated for two sets of conditions, and are discussed in some detail here. A much wider investigation was carried out into the performance of the two proposed methods, and the results are available in Appendix A. Since these mirror the results shown here in this chapter, it is not necessary to discuss them in detail. Below the simulation study reported in this chapter is described, followed by details of the simulation study shown in the appendix.

Two different models were used to compare the approximations proposed in this chapter: the F81 and F84 model. In both cases, the stationary frequencies of the nucleotides were $\pi_A = 0.1$, $\pi_i = 0.3$, $i = C, G, T$, while the transition-transversion ratio

for the F84 model was 2. Four different distances were used: 0.05, 0.4, 1.0 and 1.5. Four different sequence lengths were used: 50 bp, 150 bp, 500 bp and 1000 bp although only the results from three sequence lengths are reported. This is either because the performance of the methods for sequences of length 1000 bp is clear from the 500 bp sequences (all the approximations to confidence intervals improve with increasing sequence length), or because 50 bp is too short a sequence length for some of the conditions considered (large distances, high transition-transversion ratio).

For each set of conditions, the true confidence intervals had firstly to be evaluated. Since the distance estimator from the F81 model is a transformation of a binomial random variable, finding the true confidence intervals was straightforward. 10000 independent random variables from the appropriate binomial distribution were simulated using S-plus, and these were transformed to yield the sampling distribution of d_{F81}, and the 95% confidence intervals. To find the intervals for the F84 models, Seq-Gen (Rambaut and Grassly, 1997) was used. This is a program written to simulate a given number of data sets, consisting of sequences of a certain length according to a given phylogeny. Various models, including the F84 model, may be used for the nucleotide substitution process. 10000 data sets of two sequences for each given transition-transversion ratio, distance and sequence length were simulated. The resulting distance estimates were used to find the true confidence intervals. In most cases, equi-tailed intervals were chosen. However, where the true distribution went to infinity in the right-hand tail, the lower bound was the 5% point, while the upper bound was infinity.

For each of the models, the normal-delta and saddlepoint confidence intervals were found. Details of these calculations have been given earlier in this chapter. If the equi-tailed interval was used for the real distribution, then the intervals from the various approximations were also the equi-tailed intervals. Otherwise, the one-tailed intervals were calculated. For the F81 model, the transformed normal approximation was also calculated. Since the true values of the parameters (p or P and Q, π_i, $i = A, C, G, T$) are known, the expected intervals for each approximation may be calculated, and compared to the corresponding exact intervals. From this, conclusions about the performances of the transformed normal and saddlepoint approximations may be drawn. The resulting confidence intervals are displayed on graphs in Figures 6.1 and 6.2.

In Figure 6.1, it is seen that the transformed normal approximation is both very accurate and a considerable improvement over the normal-delta approximation for the F81 model under a wide range of conditions, including extreme cases such as large distances and short sequence lengths. For small distances (e.g., 0.05) and short sequence lengths however, its performance is comparable to the normal-delta approximation, since in these conditions, the normal approximation to the sampling distribution of a binomial probability estimator is not good ($\min\{np, n(1-p)\}$ is small). On the other

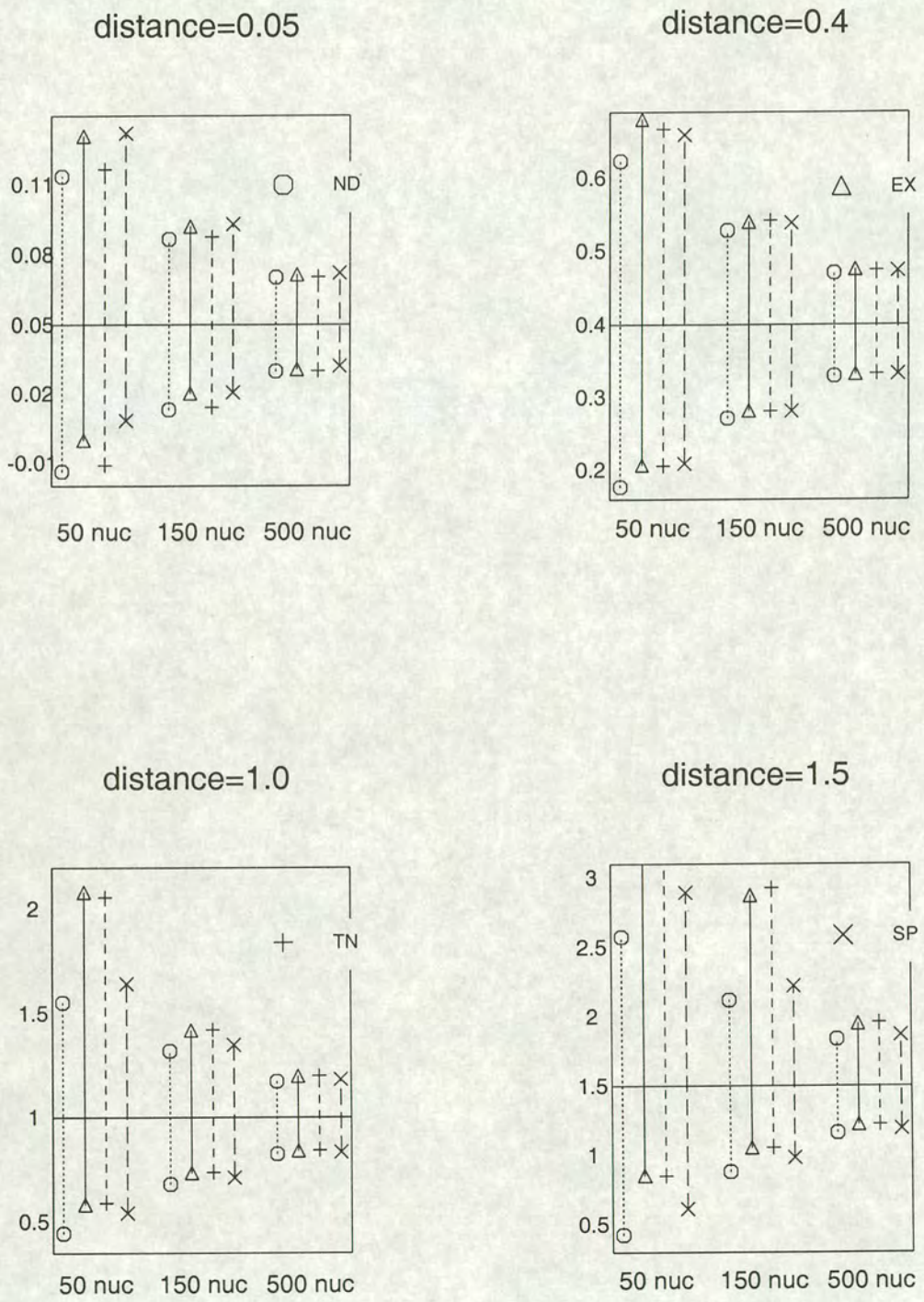


Figure 6.1: 95% confidence intervals for the F81 model for different distances and sequence lengths. ND (labelled ∘): the normal-delta approximation; EX (△): exact confidence intervals; TN (+): the transformed normal approximation; SP (×): the saddlepoint approximation. Note that lines without points at the upper end (the exact and transformed normal intervals for a distance of 1.5 and a sequence length of 50 bp) mean that the upper bound of the confidence interval is infinity.

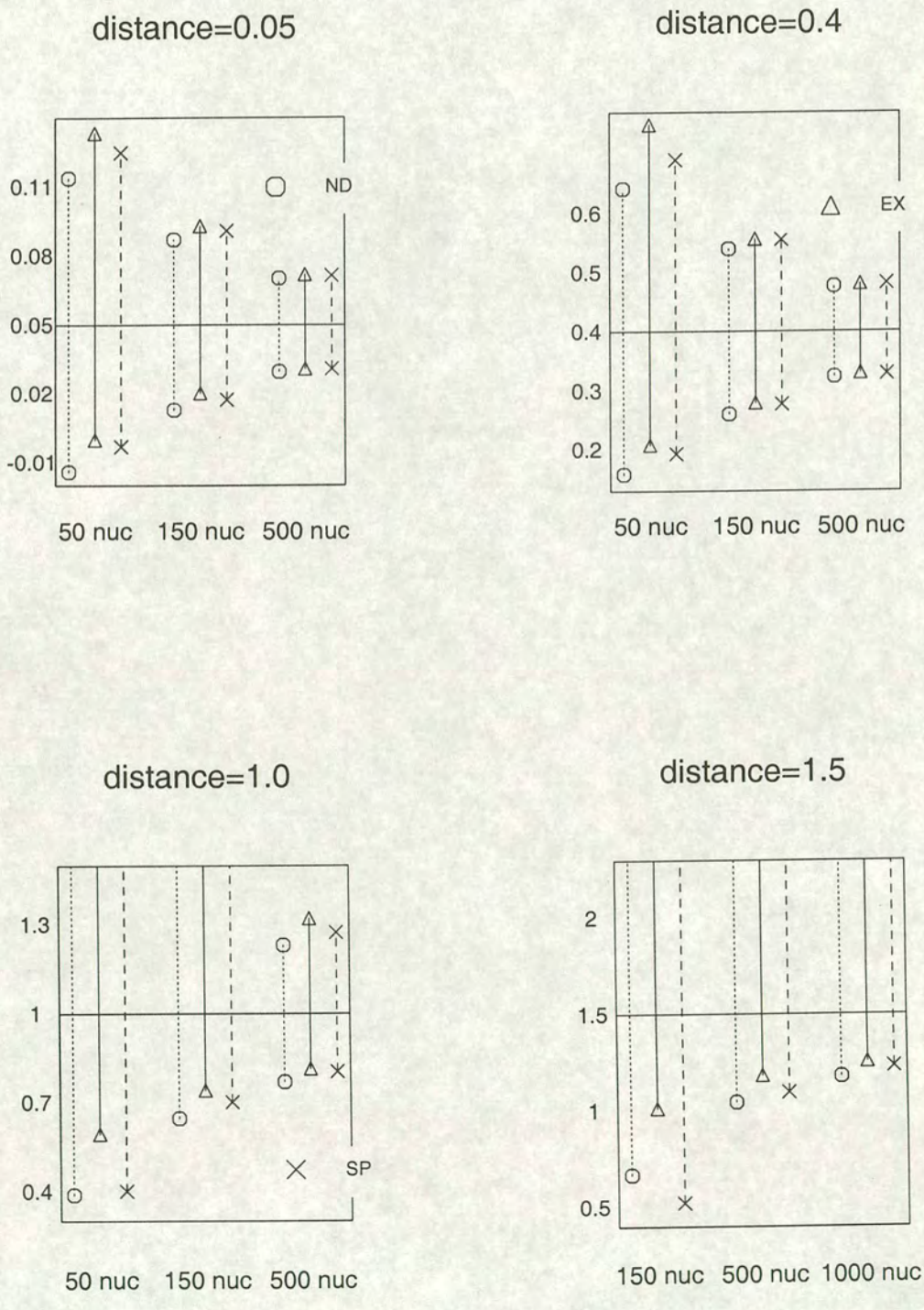


Figure 6.2: 95% confidence intervals for the F84 model, transition-transversion ratio of 2, for different distances and sequence lengths. Note that results are shown for sequences of length 150, 500 and 1000 bp when the distance is 1.5

hand, the saddlepoint approximation gives quite accurate results in this region, and is preferable to the normal-delta approximation. Overall, the saddlepoint technique gives good approximations to the true intervals, except in the extreme case of a large distance (e.g., 1.5) and a short sequence length (e.g., 50 bp). This is unsurprising as the true distribution is not very well behaved for large distances and short sequence lengths; numerical simulation of the sampling distribution of $\widehat{d}_{F81}$ often returns infinite values for the distance, caused by $\widehat{p}$ getting close to, or exceeding the value of E in (6.12).

In Figures 6.2, the saddlepoint approximation is also observed to perform better than the normal-delta approximation over a wide range of cases, especially for short sequences. Once again, the saddlepoint approximation has problems in extreme cases (a distance of 1 with a sequence length of 50; a distance of 1.5 with a sequence length of 150) resulting from the behaviour of the true distribution. In these cases its performance is comparable to, or even worse than the normal-delta approximation. The problem is more acute than for the F81 model, due to the extra parameter in the F84 model. Once the sequence length increases, however, the saddlepoint approximation quickly improves. Note that the results for 50 bp when the distance is 1.5 are omitted, since both approximations perform badly at this point due to the behaviour of the exact sampling distribution – the distance is far too large for such a short sequence. In practice, inferences will most likely be drawn from sequences separated by moderate distances; in these regions the saddlepoint approximation performs well, and should be a useful tool.

It is observed in Figures 6.1 and 6.2 that the magnitude of the difference between the normal-delta intervals and the saddlepoint or transformed normal intervals is often not very large, particularly for longer sequences. It might be thought that inferences using intervals from the two more accurate methods will not be very different from those using the normal-delta intervals. This might well be the case in some inferences, but in many cases, the accuracy of an interval is very important. Therefore, any improvement is desirable. This point will be returned to towards the end of this chapter. One further point to note is that the saddlepoint and transformed normal intervals reflect the asymmetry in the true equi-tailed confidence intervals whereas the normal-delta approximation produces symmetrical intervals.

6.7.2 Details of the extended simulation study shown in the appendix

In the initial evaluation of these approximations, a wider range of models and other conditions were examined. The results, shown again as graphs, are given in Appendix A. Both 95% and 99% intervals were calculated. Six different distances, (0.05, 0.1, 0.2, 0.4, 0.7 and 1 substitution per position) were examined. Three different sequence lengths (150, 500 and 1000 nucleotides) were used.

A range of different conditions of the F84 model were explored (six in total). Firstly, the nucleotide frequencies were varied. There were two possible sets of values:

- $\pi_A = \pi_C = \pi_G = \pi_T = 0.25$
- $\pi_A = 0.4$, $\pi_C = \pi_G = \pi_T = 0.2$

Secondly, the transition-transversion ratio was allowed to take two values:

- 0.5
- 2 (considered to be a typical value of the ratio)

Note that certain combinations of the transition-transversion ratio, and the stationary frequencies lead to simplifications of the F84 model. If the ratio is 0.5, then the F84 model reduces to the F81 model, since for the sets of nucleotide frequencies considered, $B/C = 0.5$ (B, C are calculated using equation 6.3), this being the condition for the F84 model to simplify. Furthermore, if the stationary frequencies are also equal, then the model reduces further to the JC model. Equal nucleotide frequencies with a transition-transversion ratio not equal to 0.5 will lead to the K2P model. For each set of conditions, the true confidence intervals were found as above (10000 binomial random variables or data sets were used to numerically find the sampling distribution).

Also shown Appendix A are the 99% confidence intervals for the F81 and F84 models used in this chapter to illustrate the performances of these methods. Results are shown for the 6 distances mentioned above in Figures A.1 to A.8.

As indicated above, the results are similar to those in Figures 6.1 and 6.2. The saddlepoint and transformed normal approximations appear to be a considerable improvement over the normal-delta approximation for a wide range of distances and sequence lengths. Where the transformed-normal approximation can be found (one parameter models) it is generally slightly better than the saddlepoint approximation, except for short distances. Extreme cases where the true distribution is not well-behaved continue to be a problem but that is not unexpected.

6.8 Examples using real data sets

The two approximations discussed above are now used to draw inferences from two real data sets. Firstly, the prepeptide and C-peptide encoding parts of the nucleotide sequences of human preproinsulin mRNA and rat preproinsulin-I mRNA are compared (Sures et al., 1980; Tajima and Nei, 1984). In this data set, the relative rates of change at different codon positions is of interest. A change in the first two positions of a codon often results in the amino acid encoded being changed, whereas many types of substitution at the third position leave the amino acid unaltered. Since changes in

Table 6.1: Comparisons between Rabbit and Mouse β-globin sequences using the K2P model of nucleotide substitution

	pos. 1	pos. 2	pos. 3	small	large
$\widehat{d}$	0.157	0.133	0.419	0.603	0.907
ND	0.088,0.227	0.070,0.196	0.277,0.561	0.374,0.831	0.765,1.048
SP	0.094,0.232	0.075,0.201	0.292,0.576	0.408,0.868	0.776,1.059

the amino acid encoded are often detrimental, the amount of observed change in the first two codon positions should be relatively lower than that in the third position (as sequences carrying detrimental changes tend to be removed by natural selection).

To look at the relative rates for the prepeptide and C peptide sequences, these sequences are split into the first and second codon positions (108 nucleotides) and the third position (54 nucleotides). The F81 model of nucleotide substitution is assumed. The distances are 0.190 for the first and second positions, and 0.723 for the third position, which indeed appear quite different. The 95% confidence intervals calculated using the normal-delta method are (0.098, 0.281) and (0.297, 1.149), also indicating a difference. The transformed density intervals are (0.104, 0.287) and (0.399, 1.421) for the first and second codon positions, and the third codon position respectively, which give even clearer evidence of this difference in rates.

The saddlepoint approximation may be used to give more accurate error bounds for distances when a more elaborate model of nucleotide substitution is assumed. Kimura (1980) calculated the distance and standard deviation (using the delta method) for the three codon positions using the K2P model between various mammal β-globin sequences, which may be used to establish the relative rates of nucleotide substitution between the three codon positions. Among others, he compared the rabbit and mouse sequences, and also compared the rate of evolution at the third codon position with two non-coding regions (the small introns and the large introns). Table 6.1 gives the estimated distances ($\widehat{d}$), the normal-delta (ND) and the saddlepoint (SP) 95% confidence intervals for this data.

The coding region of these β-globin sequences is 444 nucleotides long, so there are 148 nucleotides in each coding position. The small introns contain 113 nucleotides, while the large introns lead to sequences which are 557 nucleotides long (gaps in the alignment are excluded in both cases). The saddlepoint confidence intervals are somewhat different from the normal-delta confidence intervals, being asymmetrical, although in length they are equivalent to the normal-delta intervals. While the saddlepoint intervals do not alter the inferences drawn from the data (the third codon position evolves at a faster rate than the other two positions, the large introns evolve faster than the third codon position; see Kimura, 1980), they are worthwhile in that they give a more precise description of

the data. In addition, if this data were used to estimate the time since the most recent common ancestor, the saddlepoint intervals would lead to more accurate error bounds for this divergence time.

6.9 Discussion and future work

The equi-tailed confidence intervals for the models examined in this chapter tend to be asymmetrical, especially for shorter sequence lengths. The two approximations proposed here (transformed normal and saddlepoint) both exhibit that feature, whereas the commonly used normal-delta intervals do not. These approximations yield more accurate estimates of the location of the endpoints of the intervals. Hence, they are a significant improvement on the current method of confidence interval estimation.

The transformed normal approximation has limited applicability – a distance estimator must depend only on a binomial quantity, so it will be mainly restricted to the F81 and JC models. For sequences which are sufficiently diverged such that the number of transitions is near to, or has reached saturation point, it might be better to use a distance which depends on the number of transversions only. Since the sample statistic in this case (the number of transversions) is also a binomial probability estimator, the transformed normal approximation may be used. On the other hand the saddlepoint approximation has a wider range of applicability: if a distance estimator can be expressed as a simple analytical function of a sample statistic, this approximation may be used. It is, therefore, applicable to some two and three parameter models.

The accurate estimation of confidence intervals for genetic distances is very important in some applications. For example, the time since two species last shared a common ancestor is often obtained from the estimate of the number of substitutions per position separating the two species by assuming that the substitution rate per year is known. In this case, an accurate confidence interval for the distance is important to put correct error bounds on the number of years since the most recent common ancestor. Where applicable, this divergence time may also be estimated from the number of changes at the third position in a codon which do not cause the resulting amino acid to change (*synonymous* changes). If an estimator of the number of such changes may be expressed as a simple analytical function of the observed sample statistic (see, for example, Kimura, 1980), then it should be possible to derive a saddlepoint approximation to the sampling distribution of this estimator and use this to put more accurate error bounds on the time since divergence. Since rates of substitution are often very low, small changes in the confidence intervals for the distances (such as those caused by using the more accurate saddlepoint or transformed normal approximations) can have quite a large effect on the confidence interval for the time since the most recent common ancestor. The poor performance of the saddlepoint approximation for large

distances should not be a problem in such an application; large distances often mean that sequences have reached a saturation point in substitution and thus are not suitable for inferring times since divergence. Therefore, more closely related sequences should be chosen as part of the experimental design.

Andrieu et al. (1997) use interval estimation to find the exact intervals for one and two parameter models. This is quite a tedious process and, in practice, the transformed normal approximation should give comparable results over a wide range of conditions for the F81 model. The saddlepoint approximation performs well in many cases of the F84 model and provides an alternative to interval estimation. For the K2P (and F84) model, Andrieu et al. (1997) have to assume that the transition-transversion ratio is known although this will not be the case in practice. The saddlepoint approximation does not require such an assumption. In addition, the saddlepoint approximation may be more easily extended to more complicated models of nucleotide substitution.

It is well known that the models considered here lead to biased estimates in the case of short sequences and/or a high degree of divergence between the two sequences (Tajima, 1993; Rzhetsky and Nei, 1994). These authors have developed unbiased estimators for the distance between two sequences, and have estimated the sampling variance of these estimators using the delta method. However the sampling distributions of these estimators are likely to have a similar shape to that of their biased counterparts. Hence, it would be worthwhile to investigate whether the saddlepoint approximation could be applied to give better estimates of the confidence intervals for these estimators.

The models considered here are all subsets of the three parameter model proposed by Tamura and Nei (1993, see **2.5.2**), the most complicated model for which a closed form exists for the distance estimator (Yang, 1994, see **2.7.1**). Therefore, the saddlepoint approximation may also be used for this model. This approximation may also be useful for variants on the form of the distance estimator from two-parameter models. Goldstein and Pollock (1994, see **2.7.3**) derived a (closed form) additive distance estimator, LSD, which has minimal variance using generalised least squares, and found its performance to be considerably better than the K2P distance estimator. The estimator may be written as follows

$$\text{LSD} = \frac{\sigma_V^2 \widehat{S} - \sigma_{SV}^2(\widehat{S} + \widehat{V}) + \sigma_S^2 \widehat{V}}{\sigma_V^2 - 2\sigma_{SV}^2 + \sigma_S^2} \tag{6.51}$$

where

$$\widehat{S} = -\frac{1}{2}\ln[1 - 2P - Q] + \frac{1}{4}\ln[1 - 2Q]$$
$$\widehat{V} = -\frac{1}{2}\ln[1 - 2Q]$$

and

$$\sigma_S^2 = \frac{4P - 4P^2 - 16PQ + 12P^2Q + 16PQ^2 - 4P^2Q^2 + Q^3 - 4PQ^3 - Q^4}{4n(1-2P-Q)^2(1-2Q)^2}$$

$$\sigma_{SV}^2 = -\left(\frac{\alpha}{2\beta}\right)\frac{Q^2}{2n(1-2Q)^2}$$

$$\sigma_V^2 = \left(\frac{\alpha}{2\beta}\right)^2 \frac{Q(1-Q)}{n(1-2Q)^2}\,.$$

Note that $\alpha/2\beta$ is the transition-transversion ratio which is assumed known (this, of course, is a drawback but Goldstein and Pollock (1994) note that LSD is relatively robust regarding the value of the ratio used). In practice, the sample values, $\widehat{P}$ and $\widehat{Q}$ are substituted for the population values, P and Q.

While (6.51) is complicated, it still depends only on the bivariate sample statistic, $(\widehat{P}\ ,\ \widehat{Q})$. Therefore, the only differences computationally between the calculations for the K2P distance estimator and for LSD are the partial derivatives in equations (6.41) and (6.42). These may be found by hand or more easily using a computer algebra package such as MAPLE. It will then be straightforward to apply the saddlepoint approximation to LSD.

A possible extension to the F84 model and its special cases covers site-to-site rate variation. To deal with rate variation, it is assumed that the rate of evolution for some sites is faster than for others. This can be modelled using gamma mixing (Jin and Nei, 1990, see **2.7.3**). For the K2P model, they show that

$$\widehat{d}_{K2P}^{r} = \frac{a}{2}\left[(1-2\hat{P}-\hat{Q})^{-1/a} + \frac{1}{2}(1-2\hat{Q})^{-1/a} - \frac{3}{2}\right] \tag{6.52}$$

where a is the square of the inverse of the coefficient of variation of the rates within the sequences. Note that they assume that the value of a is known. Gamma mixing can also be applied to the F81 and F84 models yielding

$$\hat{d}_{F81}^{r} = Ea\left[\left(1-\frac{p}{E}\right)^{-1/a} - 1\right] \tag{6.53}$$

for the F81 model, and

$$\hat{d}_{F84}^{r} = 2Aa\left[\left(1-\frac{\hat{P}}{2A}-\frac{(A-B)\hat{Q}}{2AC}\right)^{-1/a} - 1\right] - 2(A-B-C)a\left[\left(1-\frac{Q}{2C}\right)^{-1/a} - 1\right] \tag{6.54}$$

for the F84 model. Since these are all closed form formulae, the approximations proposed in this chapter may be used.

Currently, the variance of these estimators is found using the delta method (see **6.3**), assuming the value of a is known. Thus, this estimate of the variance (which may not

be very good in the first place) only places a lower bound on the true variance, since the value of a is not known. Therefore, approximations such as the transformed normal, if applicable, and the saddlepoint can only improve inferences on the distance estimators in equations (6.52) to (6.54) above.

Improved inferences are also required for more general models (i.e., more parameters) of nucleotide substitution. Such models should reflect the true process of nucleotide substitution more closely. Since it is not possible, in general, to obtain the distance estimate as a simple function of a sample statistic, it is difficult to apply a saddlepoint approximation. The statistical properties of distance estimators from such models is another issue which requires attention.

Chapter 7

Conclusions

7.1 Summary of work

This thesis considers two problems in phylogenetics: detecting recombination in multiple sequence alignments and improving inferences from distance estimators. Several approaches have been suggested to tackle these problems.

The methodology proposed to detect recombination falls into two categories: a mainly graphical method and a statistical procedure. The graphical approach uses the *Dss* statistic to scan alignments for recombination events prior to a phylogenetic analysis. The algorithm consists of moving a window along a sequence, and calculating the *Dss* statistic for each window; changes in the topology within the window should be reflected in the value of *Dss*. Concurrent large *Dss* values suggest the presence of a recombination event. To confirm if a recombination event has occurred, the user is directed to some of the tests described in Chapter 3, although some suggestions for statistical tests based on the *Dss* statistic are given in **4.7**. An attractive feature of this method is that it can be applied to large data sets, and runs relatively quickly.

The second approach to the problem of detecting recombination considers a Bayesian model for the underlying topology at each site in a DNA alignment. If a discrete-time, first-order Markov chain is used as the prior for the topology at each site, together with the site likelihoods, the model will be structured as a Hidden Markov model. This means that certain computations (e.g., finding the maximum *a posteriori* [MAP] estimate or the renormalisation constant) are feasible. The MAP estimate consists of the sequence of topologies at each site which maximises the posterior probability; therefore it is a possible estimate of the location of recombination events. For computational reasons, only data sets of four sequences are considered here. Results from simulated and real data sets suggest this approach has potential.

To improve the estimation of error bounds for genetic distance estimators two methods were suggested. The first applies to one-parameter models of nucleotide substitution only and involves transforming normal confidence intervals to yield an almost exact re-

sult over a wide range of sequence lengths and distances. The second proposal is the saddlepoint approximation which may be applied whenever the distance estimator may be expressed in term of sums of the Transition probabilities of the Markov model of nucleotide substitution. In a simulation study, both approximations performed well in a wide range of cases.

7.2 Future work

Suggestions for future work in each of these three areas have already been given in Chapters 4, 5 and 6. A broad view of the direction which this might take is given here.

It might be possible to refine the *Dss* statistic so that it takes account of factors such as substitution rate variation and other heterogeneities in the model of nucleotide substitution along the DNA multiple sequence alignment. Consequently, any significantly high values of *Dss* would then correspond only to recombination events. If it were possible, furthermore, to find the distribution of *Dss* under the hypothesis of no recombination, then a statistical test for recombination could be implemented without having to resort to other tests for recombination. However, this is likely to be a non-trivial exercise.

There is much scope for extending and improving the Bayesian approach for detecting recombination described in Chapter 5. As indicated in **5.6**, its two main drawbacks at present are that it, firstly, only returns a point estimate of the location of a recombination event and secondly, that it is only applicable to four sequences. To deal with the former problem, Markov Chain Monte Carlo methods are proposed but require investigation. For the latter limitation, an approach using quartets, or making use of the ideas suggested by Hein (1993) are suggested. Further details are given in **5.6**.

Finding and applying the saddlepoint approximation to all possible cases is one line of investigation following on from the work described in Chapter 6. Examples of such cases are given in **6.9**. These include more complicated models of nucleotide substitution, estimators incorporating rate variation, and almost unbiased distance estimators. Improved inferences for more complicated substitution models, which do not have a closed-form for the distance estimator is another non-trivial issue.

One question which was not addressed in Chapter 6 concerns the best confidence interval to use. Should the equi-tailed interval be used, or is the equivalent highest density interval more useful (and possibly shorter), justifying the increased computational burden to find this interval? Since the saddlepoint approximation and the transformed normal approximations yield good approximations to the sampling distributions of distance estimators (albeit a continuous approximation to the discrete distribution of distance estimators from one-parameter models), these points could be investigated using these approximations.

Appendix A

Confidence Intervals for Genetic Distance Estimators – Simulation Study Results

The results for the extended simulation study mentioned in **6.7** are shown here. The exact confidence intervals for each of the six distances (0.05, 0.1, 0.2, 0.4, 0.7, 1) are plotted along with those obtained by the approximate methods (the normal-delta, the saddle-point, and where applicable, the transformed-normal approximations) for sequences of length 150, 500 and 1000 nucleotides. There are eight different figures shown:

- The F81 model ($\pi_A = 0.1$, $\pi_C = \pi_G = \pi_T = 0.3$), 99% confidence intervals;
- The JC model, 95% and 99% confidence intervals;
- The F84 model ($\pi_A = 0.1$, $\pi_C = \pi_G = \pi_T = 0.3$, transition-transversion ratio of 2), 99% confidence intervals;
- The F84 model ($\pi_A = 0.4$, $\pi_C = \pi_G = \pi_T = 0.2$, transition-transversion ratio of 2), 95% and 99% confidence intervals;
- The K2P model (transition-transversion ratio of 2), 95% and 99% confidence intervals.

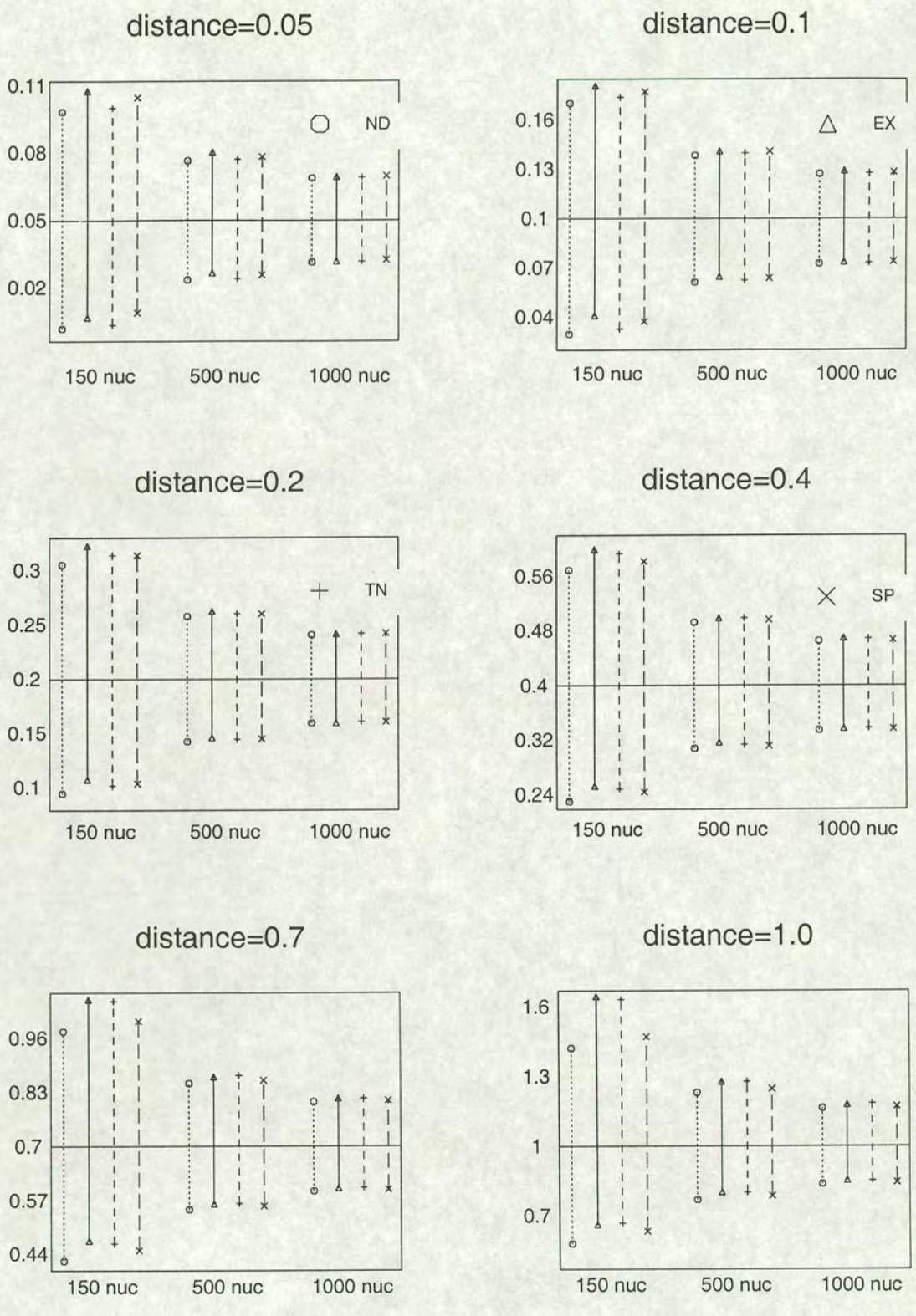


Figure A.1: 99% confidence intervals for the F81 model, $\pi_A = 0.1$, $\pi_C = \pi_G = \pi_T = 0.3$. ND (labelled $\circ$): the normal-delta approximation; EX ($\triangle$): exact confidence intervals; TN ($+$): the transformed-normal approximation; SP ($\times$): the saddlepoint approximation.

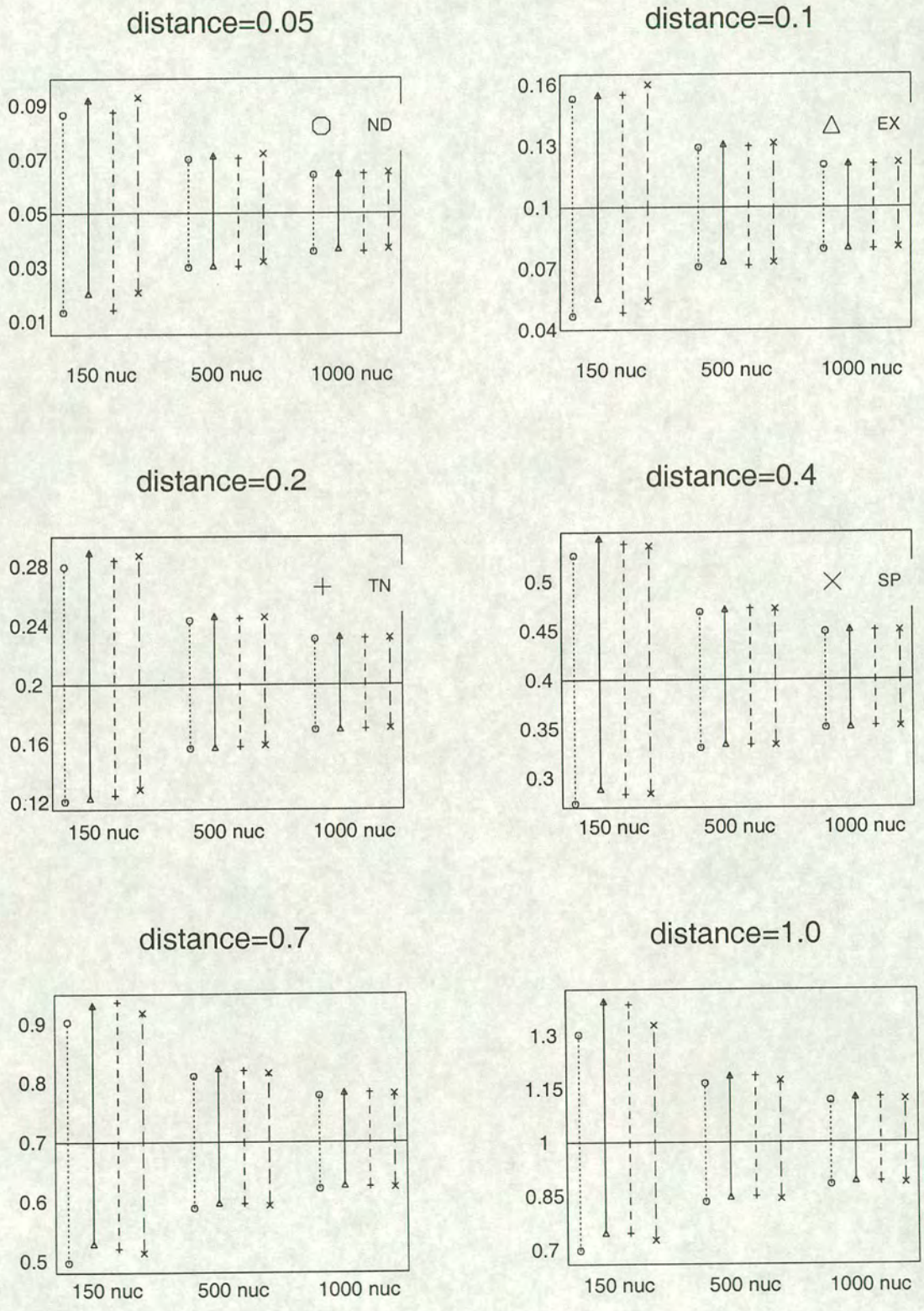


Figure A.2: 95% confidence intervals for the JC model.

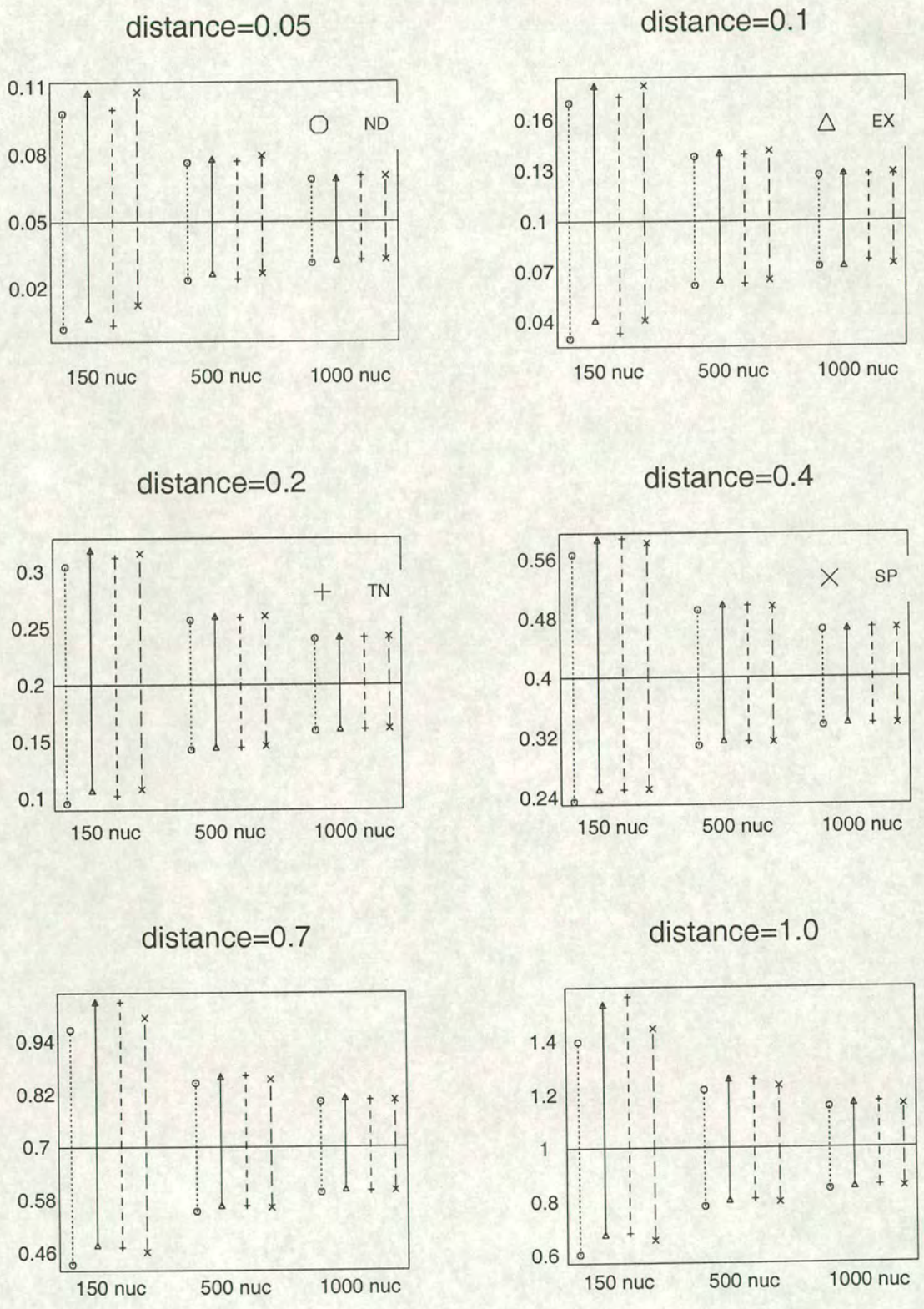


Figure A.3: 99% confidence intervals for the JC model.

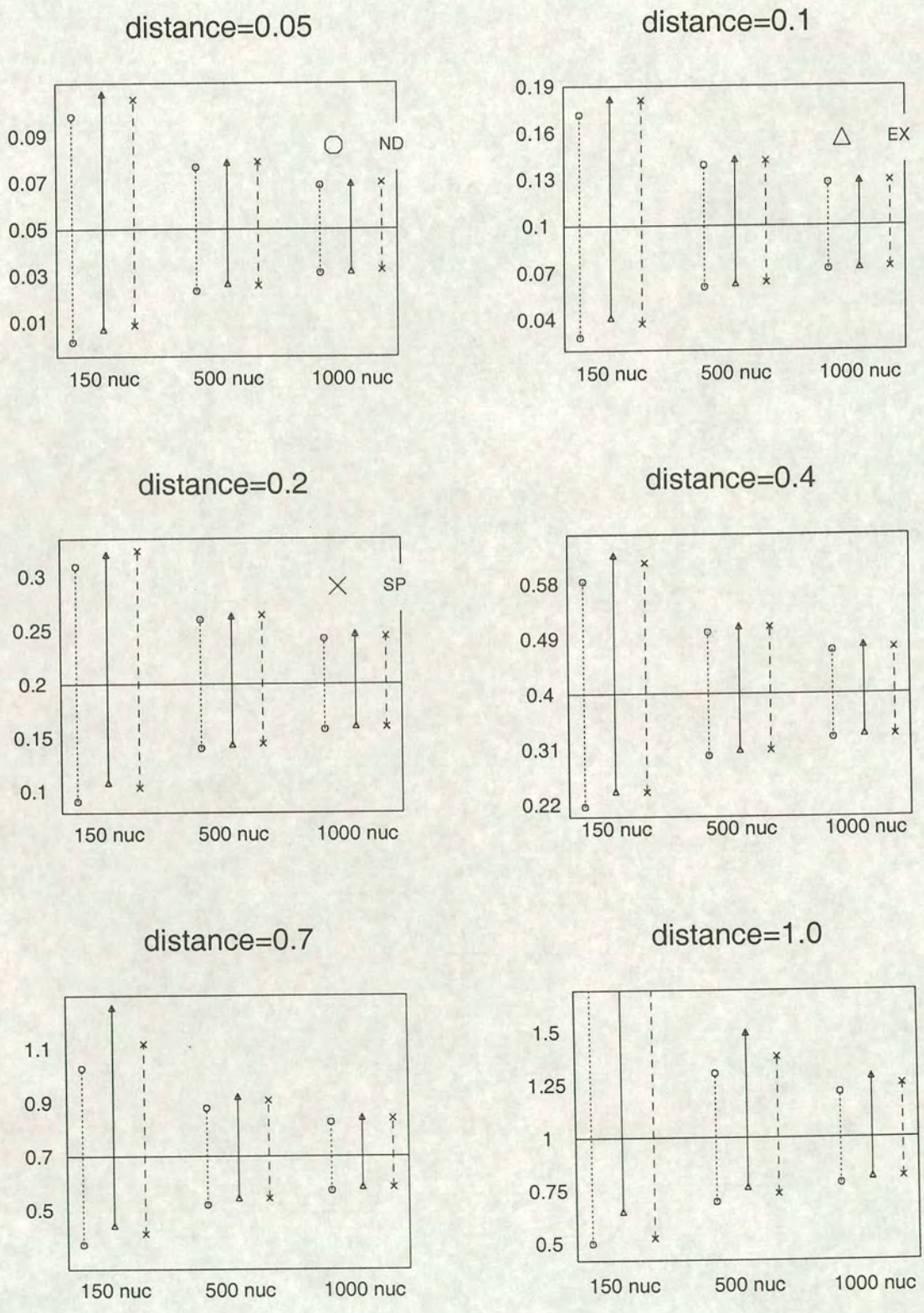


Figure A.4: 99% confidence intervals for the F84 model, $\pi_A = 0.1$, $\pi_C = \pi_G = \pi_T = 0.3$, ts/tv=2. Note that lines without points at the upper end mean that the upper bound of the confidence interval is infinity.

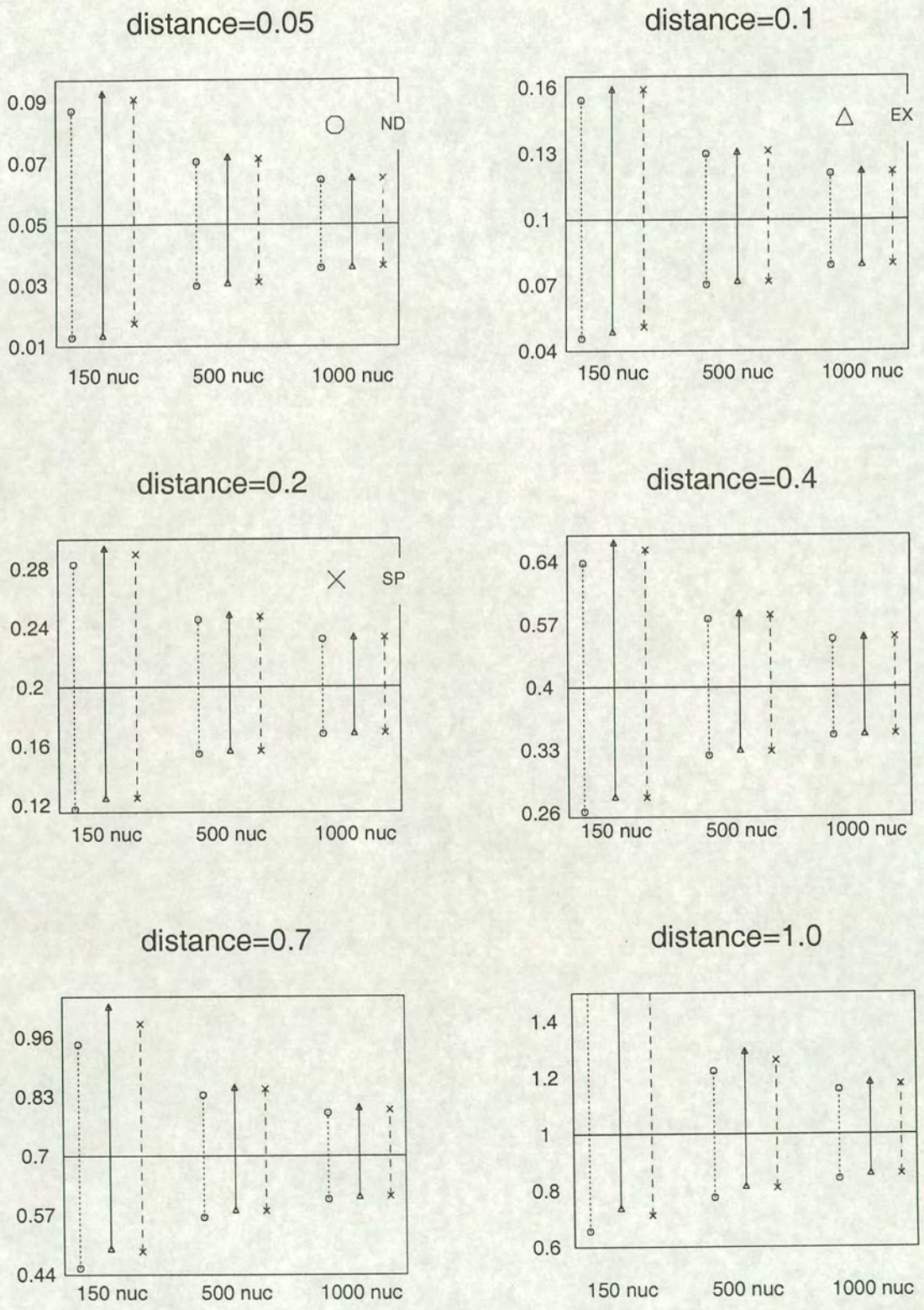


Figure A.5: 95% confidence intervals for the F84 model, $\pi_A = 0.4$, $\pi_C = \pi_G = \pi_T = 0.2$, ts/tv=2.

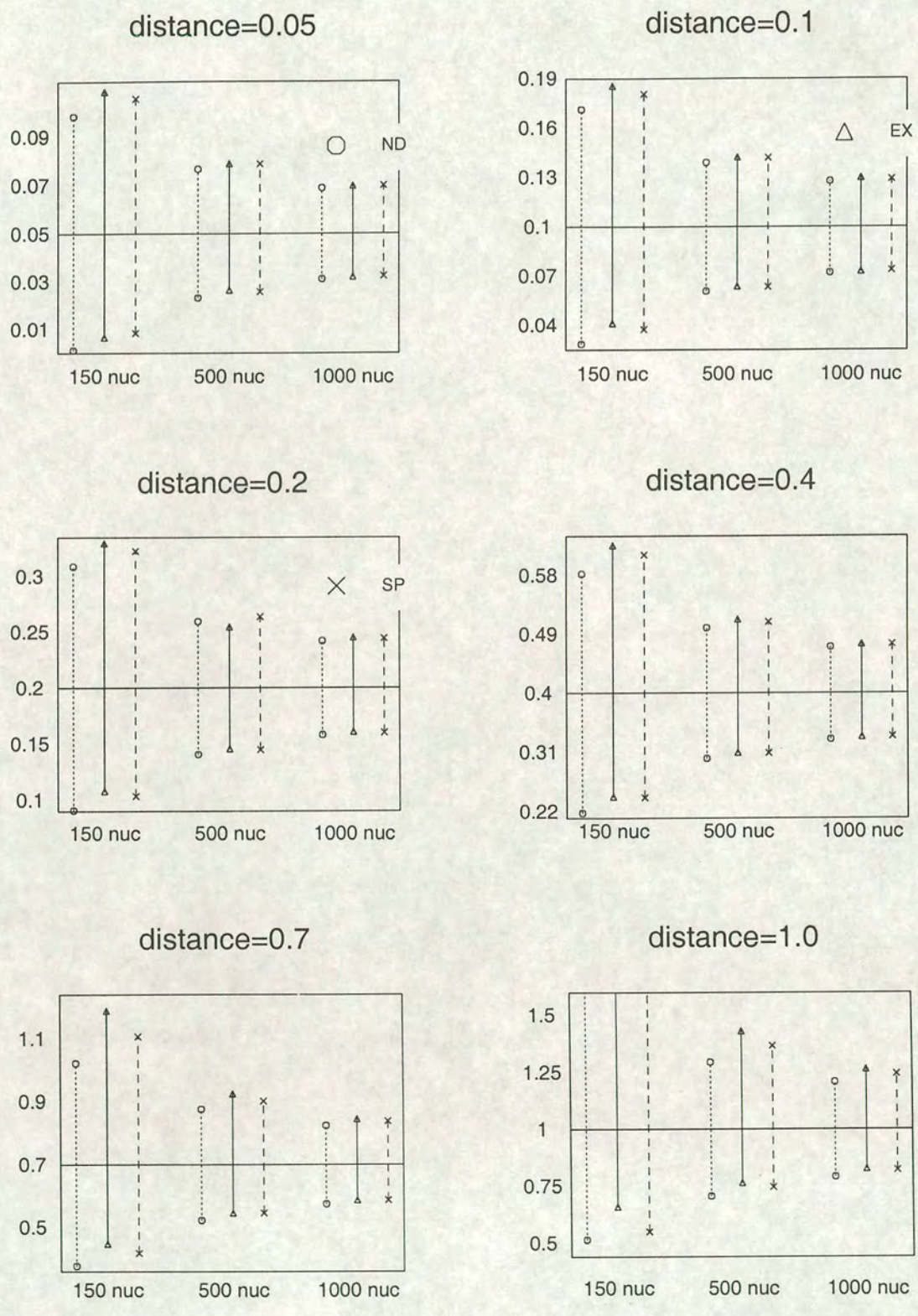


Figure A.6: 99% confidence intervals for the F84 model, $\pi_A = 0.4$, $\pi_C = \pi_G = \pi_T = 0.2$, ts/tv=2.

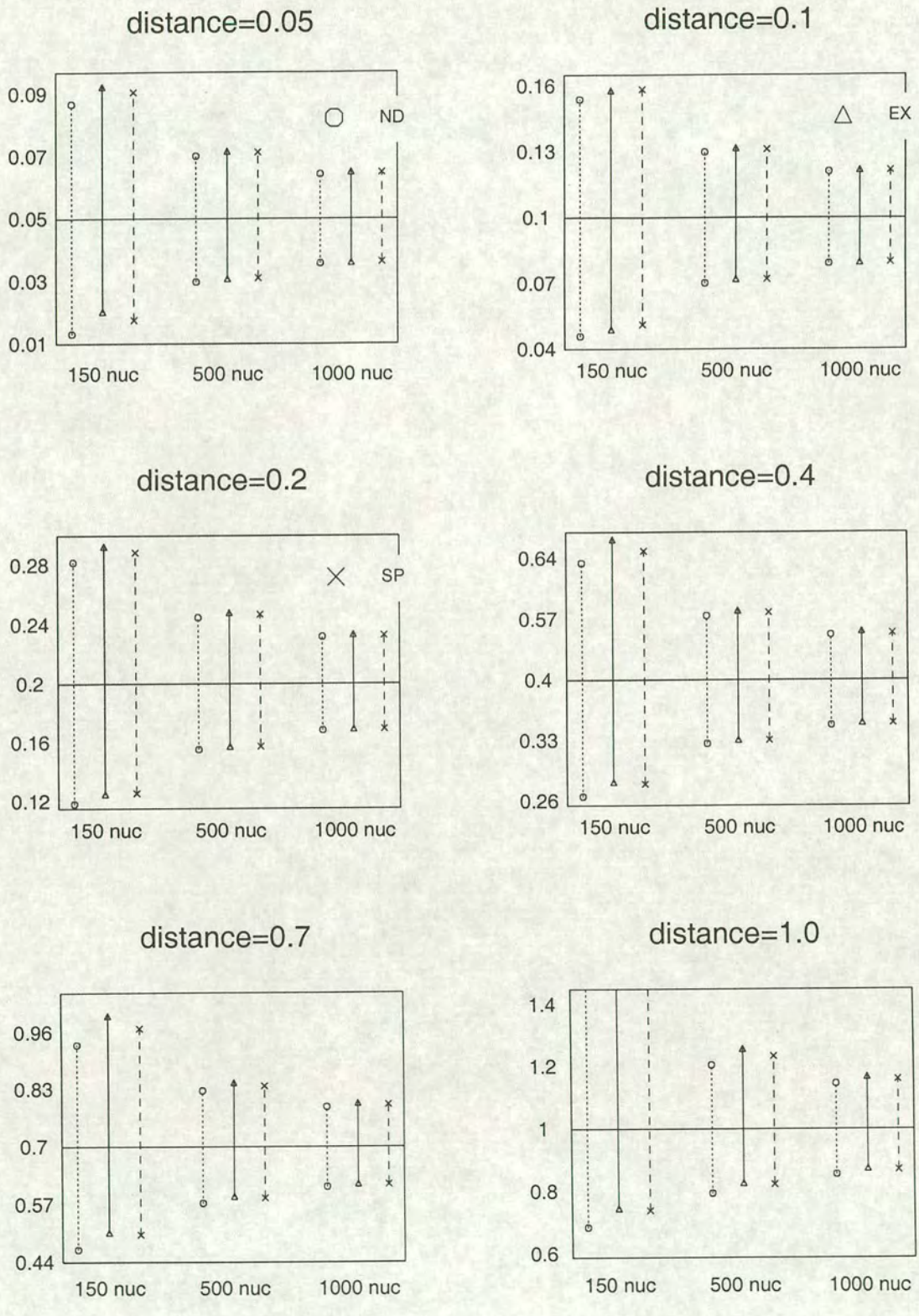


Figure A.7: 95% confidence intervals for the K2P model, ts/tv=2.

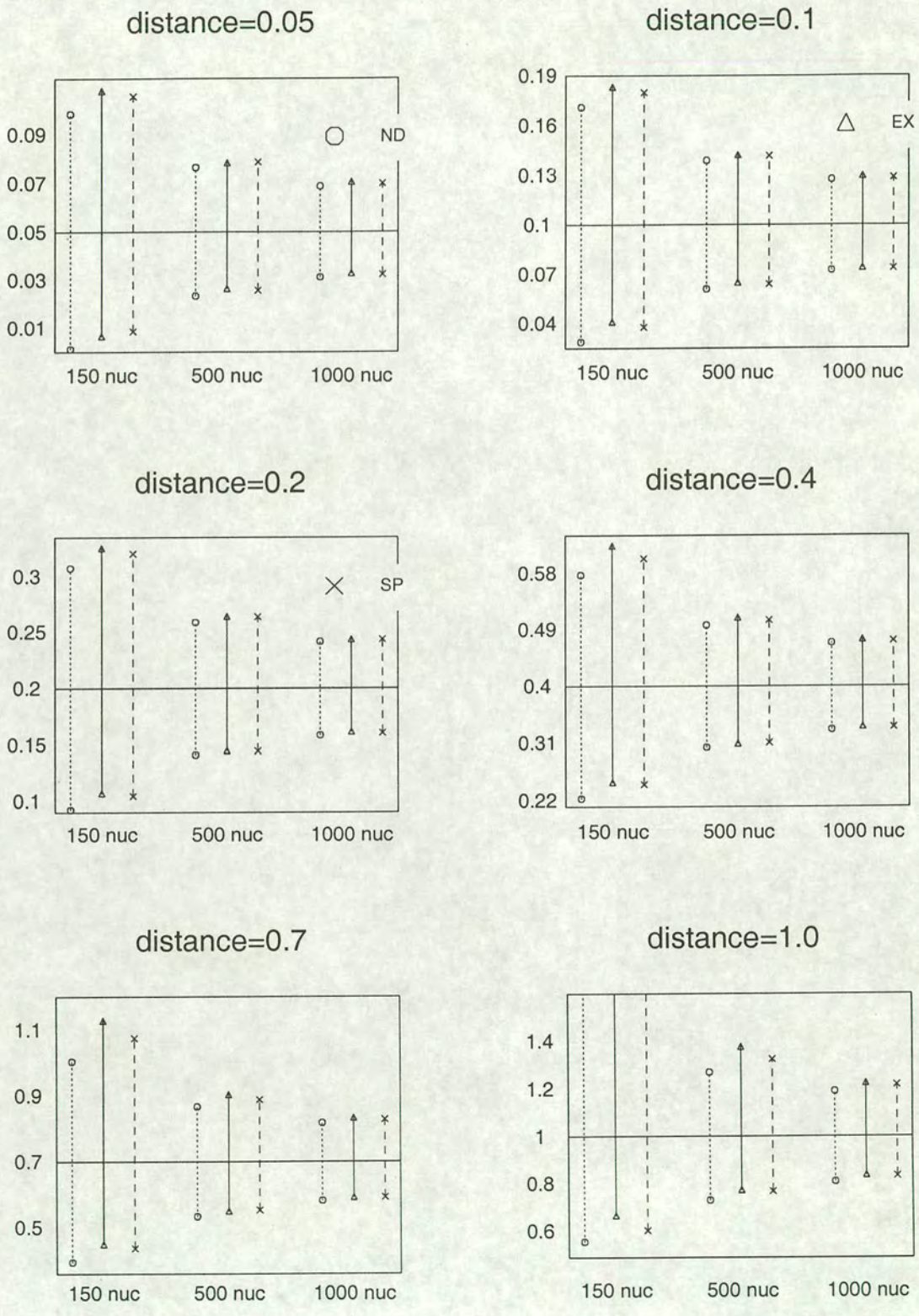


Figure A.8: 99% confidence intervals for the K2P model, ts/tv=2.

Bibliography

Andrieu, G., Caraux, G., and Gascuel, O. (1997). Confidence intervals of evolutionary distances between sequences and comparison with usual approaches including the bootstrap. *Molecular Biology and Evolution*, **14**, 875–882.

Avise, J. C. (1994). *Molecular Markers, Natural History and Evolution.* Chapman and Hall, London.

Bandelt, H. and Dress, A. W. M. (1992). Split decomposition: a new and useful approach to phylogenetic analysis of distance data. *Molecular Phylogenetics and Evolution*, **1**, 242–252.

Bollyky, P. L., Rambaut, A., Harvey, P. H., and Holmes, E. C. (1996). Recombination between sequences of Hepatitis B virus from different genotypes. *Journal of Molecular Evolution*, **42**, 97–102.

Buneman, P. (1971). The recovery of trees from measures of dissimilarity. In Hodson, F. R., Kendall, D. G., and Tautu, P., editors, *Mathematics in the Archaeological and Historical Sciences*, pages 387–395. Edinburgh University Press, Edinburgh.

Cavalli-Sforza, L. L. and Edwards, A. W. F. (1967). Phylogenetic analysis: models and estimation procedures. *Evolution*, **32**, 550–570.

Clarke, G. M. and Cooke, D. (1992). *A Basic Course in Statistics.* Edward Arnold, London, third edition.

Daniels, H. E. (1954). Saddlepoint approximations in statistics. *Annals of Mathematical Statistics*, **25**, 631–650.

Daniels, H. E. (1987). Tail probability approximations. *International Statistical Review*, **55**, 37–48.

Easton, G. S. and Ronchetti, E. (1986). General saddlepoint approximations with applications to L statistics. *Journal of the American Statistical Association*, **81**, 420–430.

Edwards, A. W. F. (1996). The origin and early development of the method of minimum evolution for the reconstruction of phylogenetic trees. *Systematic Biology*, **45**, 79–91.

Edwards, A. W. F. and Cavalli-Sforza, L. L. (1964). Reconstruction of evolutionary trees. In Heywood, W. H. and McNeill, J., editors, *Phenetic and Phylogenetic Classification*, pages 67–76. Systematics Association Publication no. 6, London.

Efron, B., Halloran, E., and Holmes, S. (1996). Bootstrap confidence levels for phylogenetic trees. *Proceedings of the National Academy of Sciences of the United States of America*, **93**, 7085–7090.

Felsenstein, J. (1978a). The number of evolutionary trees. *Systematic Zoology*, **27**, 27–33.

Felsenstein, J. (1978b). Cases in which parsimony or compatability methods will be positively misleading. *Systematic Zoology*, **27**, 401–410.

Felsenstein, J. (1981). Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of Molecular Evolution*, **17**, 368–376.

Felsenstein, J. (1985). Confidence limits on phylogenies: an approach using the bootstrap. *Evolution*, **39**, 783–791.

Felsenstein, J. (1988). Phylogenies from molecular sequences: inference and reliability. *Annual Review of Genetics*, **22**, 521–565.

Felsenstein, J. (1993). Phylip. Version 3.5c, University of Washington, Seattle. http://evolution.genetics.washington.edu/phylip.html.

Felsenstein, J. (1997). An alternating least squares approach to inferring phylogenies from pairwise distances. *Systematic Biology*, **46**, 101–111.

Felsenstein, J. and Churchill, G. A. (1996). A Hidden Markov Model approach to variation among sites in rate of evolution. *Molecular Biology and Evolution*, **13**, 93–104.

Fitch, W. M. and Margoliash, E. (1967). Construction of phylogenetic trees. *Science*, **155**, 279–284.

Gatto, R. and Ronchetti, E. (1996). General saddlepoint approximations of marginal densities and tail probabilities. *Journal of the American Statistical Association*, **91**, 666–673.

Goldman, N. (1993a). Statistical tests of models of DNA substitution. *Journal of Molecular Evolution*, **36**, 182–198.

Goldman, N. (1993b). Simple diagnostic statistical tests of models for DNA substitution. *Journal of Molecular Evolution*, **37**, 650–661.

Goldstein, D. B. and Pollock, D. D. (1994). Least squares estimation of molecular distance – noise abatement in phylogenetic reconstruction. *Theoretical Population Biology*, **45**, 219–226.

Grassly, N. C. and Holmes, E. C. (1997). A likelihood method for the detection of selection and recombination using nucleotide sequences. *Molecular Biology and Evolution*, **14**, 239–247.

Grimmett, G. R. and Stirzaker, D. R. (1992). *Probability and Random Processes*. Oxford University Press, Oxford, second edition.

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

Hein, J. (1993). A heuristic method to reconstruct the history of sequences subject to recombination. *Journal of Molecular Evolution*, **36**, 396–405.

Hillis, D. M., Mable, B. K., and Moritz, C. (1996). Applications of molecular systematics: the state of the field and a look to the future. In Hillis, D. and Moritz, C., editors, *Molecular Systematics*, pages 515–543. Sinauer Associates, Sunderland, Mass., second edition.

Huelsenbeck, J. P. (1995). Performance of phylogenetic methods in simulation. *Systematic Biology*, **44**, 17–48.

Huelsenbeck, J. P. and Bull, J. J. (1996). A likelihood ratio test to detect conflicting phylogenetic signal. *Systematic Biology*, **45**, 92–98.

Huelsenbeck, J. P., Hillis, D. M., and Jones, R. (1996). Parametric bootstrapping in molecular phylogenetics: applications and performance. In Ferraris, J. D. and Palumbi, S. R., editors, *Molecular Zoology: Advances, Strategies and Protocols*, pages 19–45. Wiley-Liss, New York.

Jin, L. and Nei, M. (1990). Limitations of the evolutionary parsimony method of phylogenetic analysis. *Molecular Biology and Evolution*, **7**, 82–102.

Juang, B. H. and Rabiner, L. R. (1991). Hidden Markov Models for speech recognition. *Technometrics*, **33**, 251–272.

Jukes, T. H. and Cantor, C. R. (1969). Evolution of protein molecules. In Munro, H. N., editor, *Mammalian Protein Metabolism*, pages 21–132. Academic Press, New York.

Kelly, C. (1994). A test of the Markovian model of DNA evolution. *Biometrics*, **50**, 653–664.

Kimura, M. (1980). A simple model for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *Journal of Molecular Evolution*, **16**, 111–120.

Kimura, M. (1981). Estimation of evolutionary differences between homologous nucleotide sequences. *Proceedings of the National Academy of Sciences of the United States of America*, **78**, 454–458.

Kimura, M. and Ohta, T. (1972). On the stochastic model for estimation of mutational distance between homologous proteins. *Journal of Molecular Evolution*, **2**, 87–90.

Kishino, H. and Hasegawa, M. (1989). Evaluation of the maximum likelihood estimate of the evolutionary tree topologies from DNA sequence data. *Journal of Molecular Evolution*, **29**, 170–179.

Kuhner, M. K. and Felsenstein, J. (1994). Simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Molecular Biology and Evolution*, **11**, 459–468.

Lake, J. A. (1994). Reconstructing evolutionary trees from DNA and protein sequences: paralinear distances. *Proceedings of the National Academy of Sciences of the United States of America*, **91**, 1455–1459.

Lanave, C., Preparata, G., Saccone, C., and Serio, G. (1984). A new method for calculating evolutionary substitution rates. *Journal of Molecular Evolution*, **20**, 86–93.

Lawrence, J. G. and Hartl, D. L. (1992). Inference of horizontal genetic transfer from molecular data: an approach using the bootstrap. *Genetics*, **131**, 753–760.

Li, W.-H. and Graur, D. (1991). *Fundamentals of Molecular Evolution.* Sinauer Associates, Sunderland, Mass.

Li, W.-H. and Gu, X. (1996). Estimating evolutionary distances between DNA sequences. *Methods in Enzymology*, **266**, 449–459.

Lockart, P. J., Steel, M. A., Hendy, M. D., and Penny, D. (1994). Recovering evolutionary trees under a more realistic model of sequence evolution. *Molecular Biology and Evolution*, **11**, 605–612.

Maynard Smith, J. (1992). Analyzing the mosaic structure of genes. *Journal of Molecular Evolution*, **34**, 126–129.

Maynard Smith, J. and Smith, N. H. (1998). Detecting recombination from gene trees. *Molecular Biology and Evolution*, **15**, 590–599.

McClure, M. A., Vasi, T. K., and Fitch, W. M. (1994). Comparative analysis of multiple protein-sequence alignment methods. *Molecular Biology and Evolution*, **11**, 571–592.

MacDonald, I. L. and Zucchini, W. (1997). *Hidden Markov and Other Models for Discrete-Valued Time Series.* Chapman and Hall, London.

McGuire, G., Prentice, M. J., and Wright, F. (1998). Improved error bounds for genetic distances from DNA sequences. Revised version submitted to *Biometrics.*

McGuire, G. and Wright, F. (1998). TOPAL: recombination detection in DNA and protein sequences. *Bioinformatics*, **14**, 219–220.

McGuire, G., Wright, F., and Prentice, M. J. (1997). A graphical method for detecting recombination in phylogenetic data sets. *Molecular Biology and Evolution*, **14**, 1125–1131.

Rambaut, A. and Grassly, N. C. (1997). Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Computer Applications in the Biosciences*, **13**, 235–238.

Robertson, D. L., Sharp, P. M., McCutchan, F. E., and Hahn, B. H. (1995). Recombination in HIV-1. *Nature*, **374**, 124–126.

Rodriguez, F., Oliver, J. L., Marin, A., and Medina, J. R. (1990). The general stochastic model of nucleotide substitution. *Journal of Theoretical Biology*, **142**, 485–501.

Rysavy, F. R., Bishop, M. J., Gibbs, G. P., and Williams, G. W. (1992). The UK Human Genome Mapping Project online computing service. *Compututer Applications in tbe Biosciences*, **8**, 149–154.

Rzhetsky, A. and Nei, M. (1994). Unbiased estimates of the number of nucleotide substitutions when substitution rate varies among different sites. *Journal of Molecular Evolution*, **38**, 295–299.

Rzhetsky, A. and Nei, M. (1995). Tests of applicability of several substitution models for DNA sequence data. *Molecular Biology and Evolution*, **12**, 131–151.

Rzhetsky, A. and Sitnikova, T. (1996). When is it safe to use an oversimplified substitution model in tree-making. *Molecular Biology and Evolution*, **13**, 1255–1265.

Saitou, N. and Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylognetic trees. *Molecular Biology and Evolution*, **4**, 406–425.

Salminen, M. O., Carr, J. K., Burke, D. S., and McCutchan, F. E. (1995). Identification of breakpoints in intergenotypic recombinants of HIV type 1 by bootscanning. *AIDS Res. Hum. Retroviruses*, **11**, 1423–1425.

Sawyer, S. (1989). Statistical tests for detecting gene conversion. *Molecular Biology and Evolution*, **6**, 526–538.

Schbath, S., Prum, B., and de Turckheim, É. (1995). Exceptional motifs in different Markov chain models for a statistical analysis of DNA sequences. *Journal of Computational Biology*, **2**, 417–437.

Schöniger, M. and von Haeseler, A. (1995). Performance of the maximum likelihood, neighbor-joining and maximum parsimony methods when sequence sites are not independent. *Systematic Biology*, **44**, 533–547.

Sneath, P. H. A. and Sokal, R. R. (1973). *Numerical Taxonomy*. W. H. Freeman and Co., San Francisco.

Steel, M. A. (1994). Recovering a tree from the Markov leaf colourations it generates under a Markov model. *Applied Mathematics Letters*, **7**, 19–23.

Stephens, J. C. (1985). Statistical methods of DNA sequence analysis: detection of intragenic recombination or gene conversion. *Molecular Biology and Evolution*, **2**, 539–556.

Strimmer, K. and von Haeseler, A. (1996). Quartet-puzzling - a quartet maximum likelihood method for reconstructing tree topologies. *Molecular Biology and Evolution*, **13**, 964–969.

Sures, I., Goeddel, D. V., Gray, A., and Ullrich, A. (1980). Nucleotide sequences of human preproinsulin complementary DNA. *Science*, **208**, 57–59.

Swofford, D. L., Olsen, G. J., Waddell, P. J., and Hillis, D. M. (1996). Phylogenetic inference. In Hillis, D. and Moritz, C., editors, *Molecular Systematics*, pages 407–514. Sinauer Associates, Sunderland, Mass., second edition.

Tajima, F. (1993). Unbiased estimation of evolutionary distance between nucleotide sequences. *Molecular Biology and Evolution*, **10**, 677–688.

Tajima, F. and Nei, M. (1982). Biases of the estimates of DNA divergence obtained by the restriction enzyme technique. *Journal of Molecular Evolution*, **18**, 115–120.

Tajima, F. and Nei, M. (1984). Estimation of evolutionary distance between nucleotide sequences. *Molecular Biology and Evolution*, **1**, 269–285.

Tamura, K. and Nei, M. (1993). Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees. *Molecular Biology and Evolution*, **10**, 512–526.

Thompson, J. D., Higgins, D. G., and Gibson, T. J. (1994). CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position specific gap penalties and weight matrix choice. *Nucleic Acids Research*, **22**, 4673–4680.

Yang, Z. (1994). Estimating the pattern of nucleotide substitution. *Journal of Molecular Evolution*, **39**, 105–111.

Zharkikh, A. (1994). Estimation of evolutionary distances between nucleotide sequences. *Journal of Molecular Evolution*, **39**, 315–329.

Zhou, J. and Spratt, B. G. (1992). Sequence diversity within the *argF*, *fbp* and *recA* genes of natural isolates of *neisseria meningitidis*: interspecies recombination within the *argf* gene. *Molecular Microbiology*, **6**, 2135–2146.