PHYLOGENETIC INFERENCE AND PATTERN CLASSIFICATION

by
Xiaofei Shi

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY AT
DALHOUSIE UNIVERSITY
HALIFAX, NOVA SCOTIA SEPTEMBER 2006

© Copyright by Xiaofei Shi, 2006
NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author’s permission.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

AVIS:
L’auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l’Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L’auteur conserve la propriété du droit d’auteur et des droits moraux qui protège cette thèse. Ni la thèse ni les extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n’y aura aucun contenu manquant.
DALHOUSIE UNIVERSITY

To comply with the Canadian Privacy Act the National Library of Canada has requested that the following pages be removed from this copy of the thesis:

Preliminary Pages
Examiners Signature Page (pii)
Dalhousie Library Copyright Agreement (piii)

Appendices
Copyright Releases (if applicable)
I dedicate this thesis to my husband and my parents.
Table of Contents

List of Tables viii

List of Figures x

Abstract xii

Acknowledgements xiii

Chapter 1 Introduction 1
  1.1 The Comparison of the Confidence Regions 2
  1.2 Testing Splits in Phylogenetic Trees 6
  1.3 Pattern Classification of Phylogeny Signals 8

Chapter 2 Likelihood 10
  2.1 Substitution Model 12
    2.1.1 DNA Sequences 12
    2.1.2 Amino Acid Sequences 14
  2.2 Accommodating Rate Heterogeneity across Sites 16
  2.3 Likelihood 17

Chapter 3 The Comparison of the Confidence Regions 20
  3.1 Methodology 21
    3.1.1 The SH Test 23
    3.1.2 The SDNB Test 25
    3.1.3 The SOWH and SSOWH Tests 26
    3.1.4 The SDPB Test 27
    3.1.5 Other Methods 27
  3.2 Simulation Methods and Analysis 30
    3.2.1 Trees 30
3.2.2 Substitution Models ........................................... 32
3.2.3 Methods .................................................. 33
3.2.4 Simulation Results ......................................... 35
3.3 Discussions ................................................... 39
  3.3.1 Comparison of the Tests ................................ 39
  3.3.2 Effects of Factors ....................................... 42
  3.3.3 Parametric versus Nonparametric Bootstrap ......... 45
3.4 Conclusions ................................................... 46

Chapter 4 Testing Splits in Phylogentic Trees 48
  4.1 The rationale of the test procedure ....................... 49
  4.2 Tree Test Methods for Individual Hypotheses ........... 52
  4.3 Procedure ................................................. 53
  4.4 Multiple P-values .......................................... 54
    4.4.1 The FDR Controlling Procedure ....................... 54
    4.4.2 The Chi-square Approximation Procedure .......... 55
    4.4.3 Hochberg’s Procedure ................................ 56
  4.5 Simulation ................................................ 57
    4.5.1 Models ............................................... 57
    4.5.2 Methods ............................................. 57
    4.5.3 Results ............................................ 58
  4.6 Real Data Analysis ........................................ 61
  4.7 Conclusion ................................................ 63

Chapter 5 Pattern Classification of Phylogeny Signals 66
  5.1 Phylogenetic Signals ...................................... 67
    5.1.1 Likelihood Weight (LW) .............................. 67
    5.1.2 P-values ............................................ 68
  5.2 Distance Measures and Gap Statistic .................... 68
    5.2.1 Distance Measures in Clustering Methods ........... 68
    5.2.2 Gap Statistic ....................................... 70
List of Tables

Table 2.1 The relationship between amino acids, three- and one-letter codes. 15

Table 3.1 The tests and their acronyms. ............................................. 23

Table 3.2 The branch lengths, the PMLTs of different sequence length (sl) and the diameters of the model trees. .......................... 32

Table 3.3 The number of trees in 95% confidence regions of SDPB, SOWH, SDNB, SSOWH, GLS, AU, SH and LW for 10-, 15- and 20-taxon trees. .................................................. 37

Table 3.4 The coverage of the confidence region with sequence length of 3000 for 10-, 15- and 20-taxon trees. ........................... 39

Table 3.5 The number of appearance of each candidate tree in 95% confidence regions for 100 data sets simulated from tree C, G and K respectively. ........................................... 44

Table 3.6 The values of the PMLT out of 100 replicates when the short branch lengths increase for the tree structure on the right in Figure 3.1. ...................................................... 45

Table 3.7 The means and the standard deviations (SD) of the cutoff values in the SDNB, SDNB with RELL and SDPB tests. .......... 46

Table 4.1 Number of errors committed when testing M null hypotheses . 55

Table 4.2 The rejection times under the constraint based on the four test methods and three multiple hypothesis testing procedures . 59

Table 4.3 The simulation result of the SDNB test and the FDR controlling procedure with increasing internal branch between subtree (8,9) and taxon 10. ...................................................... 59

Table 4.4 The number of rejections of the individual hypotheses for the 10 constraints based on combination of the AU test and the FDR controlling procedure. ................................. 63

viii
Table 5.1 The information of the underlying model trees in the two examples. 80
Table 5.2 The summary of the options in the clustering methods. 81
Table 5.3 The comparison of MEs for LWs and $P$-values as input based on different number of clusters. 86
Table 5.4 The summary of the clustering results of MEC with the top table for the first example and bottom one for the second. 88
Table 5.5 The summary of the clustering results of TSC with the top table for the first example and bottom one for the second. 91
List of Figures

Figure 1.1 Rooted (left) and unrooted (right) trees for taxon 1, 2, 3, 4 and 5 .................................................. 3

Figure 2.1 Tree 1 ........................................................................ 18

Figure 3.1 The three tree shapes with a,...,g denoting the branch lengths. ................................................. 31

Figure 3.2 The structures of 10-, 15- and 20-taxon trees for simulation ......................................................... 33

Figure 3.3 The simulation results of 5-taxon trees. ......................................................................................... 36

Figure 3.4 The 15 possible tree topologies of 5 taxa .................................................................................. 43

Figure 4.1 A 10-taxon tree with structure as (1: 0.016185, 2: 0.019150, (3: 0.032104, ((4: 0.016474, 5: 0.020743): 0.034872, (6: 0.049471, (7: 0.063597, ((8: 0.024445, 9: 0.022501): 0.003587, 10: 0.026746): 0.036609): 0.005): 0.010505): 0.002312): 0.012872) ......... 49

Figure 4.2 An example for demonstrating the basic idea of the constraint test. .............................................. 50

Figure 4.3 The structure of the tested tree under $H_0$, with taxon 1 and 2 from inside of the split and the other two from outside of the split. ................................................................. 51

Figure 4.4 The ML tree of 46 taxa. ................................................................................................................. 65

Figure 5.1 Comparison of the adapted gap statistic and the gap statistic on estimating the number of groups. .................................................................................................................. 78

Figure 5.2 An example of the standard heatmap and the color palette of the corresponding LWs. ................. 82

Figure 5.3 The heatmap of LWs (top) and $P$-values (bottom) of the two examples. ........................................ 85

Figure 5.4 The heatmaps of the hierarchical method for the LWs (top) and $P$-values (bottom). ....................... 87
Figure 5.5  The heatmaps of MEC with the left one for the first and the right for the second examples. ......................... 89
Figure 5.6  Demonstrations of adopted gap statistic, heatmap and gap statistic for TSC on the two simulated data sets. ................. 90
Figure 5.7  Demonstrations of the heatmaps of MSE and TSC, and adopted gap statistic and gap statistic for TSC on the real data .... 95
Abstract

We focus our attention on problems related to phylogenetic trees which present the evolutionary history of life. It is composed by three parts, the first two for phylogenetic inferences and the third one for pattern classification.

In the first part, six existing and two new proposed procedures for constructing confidence regions of the true evolutionary tree are evaluated both in terms of coverage and size. We compare them on simulated trees and give reference about their performances under different conditions.

Then we develop a new method for testing a portion of a tree (called a split) based on multiple tests of many 4-taxon trees in the second part. This is particularly useful when the phylogenetic tree constructed by other methods has a split which is difficult to explain from a biological point of view. A combination of the test methods and the methods dealing with the P-values of these 4-taxon trees is recommended, which provides strong power and reasonable familywise error rate.

The last section is about clstering genes based on their phylogenetic signals. Evolutionary trees can be constructed from a number of different genes. But different genes may have different underlying evolutionary histories. We propose two clustering methods to obtain the subsets of the genes with the same or similar evolutionary process, and compare their performance with a classical clustering method based on simulated data.
Acknowledgements

I would like to express my gratitude to Dr. Chris Field and Dr. Hong Gu, without whose guidance and encouragement this work would not have been achieved. I would like to thank my husband, Xin Liao, my parents and all my friends for their support that helped me to walk through the hard days. I also extend thanks to Dr. Ed Susko, Dr. Joseph Bielawski and Dr. Brian Golding for reviewing this thesis. Thank you very much.
Chapter 1

Introduction

How the evolutionary history of life can be inferred is called the phylogeny problem. Cavalli-Sforza and Edwards (1967) made an important contribution to phylogenetics by the early realization that the phylogeny problem was a statistical inference problem.

Inferring a phylogeny is an estimation procedure to make a "best estimate" of an evolutionary history based on the incomplete information contained in molecular data, such as DNA and amino acid sequences. In molecular evolution, the evolutionary history is represented by trees and the "best estimate" is a tree topology most consistent with the observations. However, the best tree based on some criteria may not represent the true evolutionary history because of insufficient information or errors. To construct the confidence region of the true tree is one way to represent this uncertainty. Methods based on the maximum likelihood approach and the distance approach are very popular to select one or more preferred trees among a set of possible phylogenies. Interestingly, a number of the existing confidence region methods can give rather different results. How to evaluate their performance is our first problem.

When part of the phylogeny is known from prior knowledge or biological evidence, it is possible that the ML tree conflicts with it. In this situation, it is of interest to make inference about a prespecified split in a tree instead of the whole tree structure. The use of bootstraps was suggested to solve this problem by Felsenstein (1985) and improved by Efron et al. (1996). We address this issue as a hypothesis testing problem and implement the standard test methods on it by decomposing the original null hypothesis with partially unspecified topology to many hypotheses with fixed null hypothesis.

Whereas the first two problems in this thesis address issues in phylogenetic inference, the last problem is to classify genes based on their phylogenetic signals.
Evolutionary trees can be constructed from a number of different genes to express the evolutionary process of species. However different genes may have different underlying evolutionary histories. Clustering methods help us to obtain the subsets of the genes with the same or similar evolutionary process.

The structure of this thesis is arranged as below. Since the likelihood is essential for all three topics, how to calculate the likelihood for DNA and amino acids sequences, along with the relative parameters, such as substitution models and rate heterogeneity, are introduced in chapter 2. The comparison of the confidence regions is presented in chapter 3, followed by testing a split of phylogenetic trees which involves multiple comparisons in chapter 4. Some test methods mentioned in chapter 3 are applied in chapter 4 as well. Then the methods for pattern classification of phylogeny signals are given in chapter 5. This chapter is not highly related to the previous two chapters, but still tightly tied to likelihoods of the trees. Finally future research directions are discussed in chapter 6.

More details of the problems discussed in chapter 3, 4 and 5 are given respectively in section 1.1, 1.2 and 1.3, composing the rest of this chapter.

1.1 The Comparison of the Confidence Regions

Trees are the common structures to express an evolutionary history in phylogeny. According to the book *Molecular Evolution - A Phylogenetic Approach* (Page and Holmes 2000), some explanations of the terminology about phylogenetic trees are given below:

A *tree* is a mathematical structure which is used to model the actual evolutionary history of a group of sequences or organisms. This actual pattern of historical relationships is the *phylogeny* or *evolutionary tree* which we try to estimate. A tree consists of *nodes* connected by *branches*.

*Terminal nodes* represent sequences or organisms for which we have data; they may be either extant or extinct.

*Internal nodes* represent hypothetical ancestors; the ancestor of all the sequences that comprise the tree is the *root* of the tree.
Figure 1.1: Rooted (left) and unrooted (right) trees for taxon 1, 2, 3, 4 and 5.

A rooted tree has a node identified as the root from which ultimately all other nodes descend. Hence a rooted tree has direction.

A unrooted tree lacks a root. Hence it does not specify evolutionary relationships in quite the same way, and it does not allow us to distinguish ancestors and descendants.

In phylogenetic analysis, the first step usually involves the estimation of the trees based on the changes in DNA or amino acid sequences among species. While the changes among species tell the evolutionary history, the unchanged information doesn’t. During phylogenetic analysis, only the information at the terminal nodes can be observed. Since there are 4 (for DNA) or 20 (for amino acid) states for a single site of an internal node and a lot of state combinations for more than one internal nodes, choosing one of states for a internal node would be unlikely to be correct. For example, an unrooted tree of 5 species has three interior nodes, so that at each site there are \(4^3 = 64\) possible assignments of nucleotides to the interior nodes. A method that assigns nucleotide types to the interior nodes at random would have a possibility 0.0156 of being correct if all assignments were equally likely. As the number of interior nodes grows, it is less likely that we would make the correct assignment to the internal nodes. The maximum likelihood approach (Cavalli-Sforza and Edwards 1967; Felsenstein 1981) and a distance approach (Cavalli-Sforza and Edwards 1967)
deal with the unknown information of interior nodes from different points of view.

The maximum likelihood estimation of phylogeny was discussed at the 1964 meeting of the Systematics Association and at the Cold Spring Harbor Symposium on Quantitative Biology. Neyman (1971) applied the method of maximum likelihood to analyze nucleotide sequence data for three species using a simple probabilistic model with symmetric changes similar to the Jukes-Cantor formula (Jukes and Cantor 1969). Felsenstein (1973, 1981) further developed the idea by proposing a maximum likelihood method for inferring evolutionary trees using discrete nucleotides of extant species. During the likelihood calculation, the states of the interior nodes were regarded as random variables. By considering all the possible states of interior nodes with the corresponding probabilities, the likelihood function carries the observed and unobserved information. The tree with the maximum likelihood is considered as the best estimate of the evolutionary history.

While likelihood approaches attempt to estimate the actual number of changes according to an evolutionary model, the distance approaches use corrected distances that account for superimposed changes (Cavalli-Sforza and Edwards 1967). The corrected distances are estimates of the true evolutionary distances, which reflect the mean number of changes per site that have occurred between a pair of sequences. If the true evolutionary distance can be determined by the differences between each pair of observed sequences under study, it would be equal to the sum of the lengths of each branch lying on the path between the members of each pair. This idea expands to fit the distances to an unrooted tree such that all pairwise distances are equal to the sum of the branch lengths along the path connecting the corresponding taxa. Unfortunately, due to the finite amount of available data, random errors will cause deviation of the estimated evolutionary distances from the actual distances even when evolution proceeds exactly according to the model used for distance correction. Many methods have been described that derive a tree and an associated set of branch lengths that comes closest (in some sense) as the estimation for a matrix of pairwise distances. These methods typically attempt to optimize an objective function that quantifies the degree of disagreement between the path lengths and the observed distances. The tree with the smallest disagreement can be treated as the best estimate.
By realizing that the best tree may not represent the true evolutionary process, the confidence region methods of phylogenetic trees based on the two approaches are considered. Here we consider the confidence region methods on the basis of tests of hypothesis.

Maximum likelihood method is applied in many tests. Kishino and Hasegawa (1989) developed a method for testing two topologies, referred as the KH test. The KH test was designed for comparing two topologies but is often used for comparing many topologies. This use of the KH test leads to over confidence for wrong trees, because the sampling error due to the selection of the maximum likelihood topology is overlooked in it. Swofford et al. (1996) observed that the KH test should be applied only when the trees in question were specified with a prior. Shimodaira and Hasegawa (1999) made the same point and described a correct nonparametric version for the case in which the ML tree is one of the tested topologies. The test is called the SH test. However its hypothesis includes a group of trees rather than trees independently, which makes it conservative. Also the P-values increase with increasing number of candidate trees as mentioned by many authors (e.g. Strimmer and Rambaut 2001). In 2000, Goldman et al. summarized the published test methods. Besides the KH and SH test, the SOWH test was also mentioned, which is named after the authors who originally described it (Swofford et al. 1996; Hillis, Mable and Moritz 1996). Unlike its predecessors, the distribution of its test statistic is constructed by parametric bootstraps. But it is not widely used since many results have shown that its confidence region only contained the ML tree when it was applied on real data even with high confidence level (e.g. Goldman, Anderson and Rodrigo 2000; Buckley 2002). Antezana (2003) proposed a star version of the parametric bootstrap to alleviate the apparent overconfident results of the SOWH test. Strimmer and Rambaut (2001) constructed the confidence region of evolutionary trees based on likelihood weight which was the proportion of the likelihood of a tree over the sum of the likelihoods of all candidate trees. The approximately unbiased (AU) test of Shimodaira (2002) adjusts the bootstrap probability for possible curvatures to give second order accurate P-values. We propose two new methods according to the bootstrap paradigm, in which the distribution of the test statistic are constructed not under the tested tree,
but the ML tree of the observation. The difference between the two methods is the use of the bootstrap: nonparametric and parametric, respectively.

In distance approaches, the negative side of reducing character data to pairwise distances is the information lost during this transformation. Thus, following Cavalli-Sforza and Edwards (1967), we view distance methods as less desirable approaches than full maximum likelihood ones. However, the distance methods provide an alternative solution by their fast speed when likelihood methods are infeasible, such as analyzing large data sets. The generalized least squares (GLS) test is the only distance method considered in this thesis (Susko 2003). The use of generalized least squares in phylogeny was mentioned by Cavalli-Sforza and Edwards (1967) and considered in more detail in Bulmer (1991). Susko (2003) extended the formula which gave the covariances of the pairwise distance and made it applicable to most of the maximum likelihood distances. The asymptotic chi-square distribution of its test statistic is used to calculate the $P$-value.

In chapter 3, eight confidence region methods mentioned above are evaluated both in terms of coverage and size of the confidence regions. The data sets are simulated from trees with small and large number of taxa. A good method which provides small size and reasonable coverage will be recommended.

1.2 Testing Splits in Phylogenetic Trees

In the estimation of phylogenetic trees, it is often of interest to know whether certain species are together to form a split, which is a partition of the leaves by removing one edge of the tree. In other words, the interest is in testing some partial structure of a tree. In this case, the null hypothesis would only specify part of the tree structure. It means that $H_0$ doesn't specify the tree structure of this split and the remaining species, and we have to deal with the issue of nuisance parameters in carrying out the test.

We were motivated in this study by the example in Andersson and Roger (2000) where the ML tree of 46 eubacterial and eukaryotic homologs was constructed. The eukaryote species were placed into two distinct splits with five exceptions. Andersson and Roger were interested in testing whether any of the five exceptional eukaryotes
could belong to one of the two eukaryote splits.

Currently, most of the existing test methods are designed to test a fully fixed tree structure, such as the methods mentioned in section 1.1. It is possible to use these tests by fixing the unspecified part of the tree at the maximum likelihood values obtained under the null hypothesis that is called the constraint ML tree. Andersson and Roger (2002) used this approach and applied SH and SOWH to test the constraint ML tree as to test their hypotheses about the five exceptional eukaryotes. The $P$-values varied extensively in the SH test, depending on which additional trees were included in the test set of trees and typically exceeded 0.75. As the most conservative method, the SH test did not reject any tested topologies, whereas the SOWH test rejected all tested topologies with very small $P$-values. Even ignoring the issues of performances of the test methods, the application of using the constrained ML tree as the tested one may be inappropriate, since the original hypothesis is replaced by the constraint ML tree. The rejection of the hypothesis may be due to the null hypothesis itself or the other part of the constrained ML tree. In testing that any of these five eukaryotes is in the same split as one of the two eukaryotic groups, only a partially fixed structure should be tested, not the concrete tree topology.

The use of bootstrap selection probability to assign the confidence level for constraints was mentioned by Felsenstein (1985). The bootstrap selection probability is assessed as the “confidence” of each split of an observed tree, based on the proportion of bootstrap trees showing the same split. It was corrected to better agree with the standard ideas of confidence levels and hypothesis testings by Efron et al. (1996) by considering the curvature of the boundary of the trees in order to have second order accuracy.

In chapter 4, we develop a new method for testing a portion of a tree (called a split) based on multiple tests of many 4-taxon trees. This is particularly useful when the phylogenetic tree constructed by other methods has a split which is difficult to explain from a biological point of view. Our approach will be to decompose the original hypothesis into a number of individual hypotheses, in which the tested trees have fixed structures due to the original hypothesis. Some existing test methods can then be adopted for these hypotheses to find $P$-values. A conclusion about the original
hypothesis can be made through dealing with these $P$-values. The combination of the test methods and the methods dealing with $P$-values will be recommended which provides strong power and reasonable familywise error rate.

1.3 Pattern Classification of Phylogeny Signals

Traditional sequence-based phylogeny relies on gradual sequence changes over time. There are three main problems with using single genes to determine the relationships between species: insufficient number of informative sites, variability of evolutionary rates in different taxa and effect of lateral gene transfer. The first two factors add uncertainty to tree constructions; the last factor leads to protein phylogenies being genuinely different from the species phylogeny. In order to overcome these problems, we can concatenate many sequence alignments into one and use the combined long sequence for tree construction. Biologists often have looked for indications that it is safe to concatenate sequence data sets to get longer sequences, and a number of statistical tests exist for determining whether such a combination is "safe". Roughly speaking, these tests seek to determine whether the two data sets come from the same underlying evolutionary tree. While such test methods generally involve multiple comparisons and heavy computational burden, here we attempt to find the evidence for concatenation by a data mining tool: clustering.

The term cluster analysis encompasses a number of different algorithms and methods for grouping objects of similar kind into respective clusters. Its process is to partition the input data into subsets or clusters such that objects in the same cluster are more closely related to one another than objects assigned to different clusters.

Ideally, the genes with the same evolutionary history are assigned into the same cluster by the methods. Since the evolutionary history is represented by phylogenetic trees, it is natural and simple to assign the genes with the same "best" or ML trees into the same cluster. However as we know, the ML tree may not be the true tree for many reasons, such as observation errors in sequences and model misspecification for calculating likelihoods. It is more reasonable to consider all the trees which are possible to have the true structure. As the inputs for the cluster methods in chapter 5, the likelihood weights are considered as the probabilities in a multinomial distribution.
It means that we need to cluster the distributions of the true tree over genes, which makes this problem special.

A large number of clustering algorithms have been proposed with hierarchical clustering and \(k\)\text{-}means (MarQueen 1967) as the most popular ones. In hierarchical clustering, a nested set of clusters is created by combining the two closest objects together at each level. At the lowest level, each object is in its own unique cluster. At the highest level, all objects belong to one big cluster. In \(k\)\text{-}means algorithm, a heuristic search is used to assign objects to different clusters such that the sum of squared Euclidean distances between the objects within a cluster or the center of the cluster is minimized.

We propose two new methods for clustering genes as alternatives to the existing methods. The two-stage clustering (TSC) method is an advanced version of hierarchical clustering. And similar to the hierarchical clustering, the TSC method creates nested clusters by combining two closest objects together in its second stage. But in the first stage, the genes are divided into subgroups based on their order given by singular value decomposition (SVD) to minimize the within-cluster distance. So the initial objects in the second stage are not single genes but groups of genes. The first stage doubles the assurance that the tightly clustered genes won't be assigned into different clusters by chance. While TSC can be generally implemented on many clustering problems, the minimum entropy clustering (MEC) method is particularly designed for clustering genes, in which the inputs are the probability distributions of the true evolutionary trees of different genes. Entropy measures the uncertainty of a probability distribution. For phylogeny signals, it is small when one tree has a high probability being true and large when many trees have comparable probabilities. Since the concatenated sequences should provide a stronger signal of the true tree than any of the single sequences if they have the same evolutionary history, the MEC method puts genes into the same cluster when the concatenated sequences have smaller entropy. No prior information of the number of clusters is needed for this method. In other words, the method provides the estimate of the number of clusters automatically, which is a major advantage over other methods.
Chapter 2

Likelihood

To deal with the three topics in this thesis, likelihoods are always involved. The likelihood of a phylogenetic tree is proportional to the probability of observing the sequence data for the given tree. In other words, the likelihood can be calculated for any given tree topology.

For calculating likelihoods, the evolutionary process is assumed to be a Markov chain specified by a substitution matrix (or instantaneous rate matrix) $Q$ indicating how the sequence changes along a tree. For mathematical simplicity and ease of computation, extra restrictions have normally been placed on the structure of $Q$, leading to various parametric models. For DNA sequences, the simplest model is JC (Jukes and Cantor 1969) model in which all the changes among the four nucleotides are assumed to occur with equal probability. Kimura’s (1980) model (K80) allows transitions and transversions to occur with different rates, while Felsenstein’s (1981) model (F81) allows the four nucleotides to have unequal frequencies at equilibrium. Note that transitions occur when a purine nucleotide ($a$ and $g$) is substituted for another purine, or a pyrimidine ($c$ and $t$) is replaced by another pyrimidine; transversions occur when a pyrimidine is substituted for a purine. The model of Hasegawa et al. (1985) (HKY) merits special attention as it allows both different rates for transitions and transversions and different nucleotide frequencies, which is the natural extension to all the above three. Felsenstein further developed the HKY model by considering the different frequencies of pyrimidines and purines in the F84 model (Felsenstein and Chruchill 1996). The most popular general time-reversible (GTR) model allows more parameters in $Q$ matrix and all the above models are nested in it.

While many models of nucleotide substitutions have been proposed, relatively few attempts have been made to refine the substitution models of amino acid. Early models of protein evolution were limited by the computational issues associated with
likelihood inference. To overcome these limitations, empirical methods were devised to approximate the substitution matrix by counting the number of inferred substitutions. The first widely used models of protein evolution is the Dayhoff model (Dayhoff et al. 1987), representing data from 1572 counted substitutions. The JTT model (Jones et al. 1992) is an updated version of Dayhoff model, which involves more sequences but still follows the similar principle of empirical methods.

At the early stage of likelihood implementation, it was assumed that every site evolved at the same rate. Violation of this assumption can have devastating consequences. There have been many attempts to account for such rate variation in phylogenetic analysis. One approach assumes that rates over sites are random variables drawn from a continuous distribution with the gamma distribution being commonly used (e.g. Nei and Gojobori 1986; Jin and Nei 1990; Tamura and Nei 1993). The gamma distribution was first incorporated into a likelihood analysis by Yang (1993). However his algorithm involves too much intensive computation to be popular. As a result, though a continuous distribution may seem to be more reasonable biologically, most approaches divide the sites into several categories with different rates. The simplest model of this sort assumes that a proportion of sites are invariable while others are changing at a constant rate (e.g., Palumbi 1989; Hasegawa and Horai 1991). In 1994, Yang combined the two approaches to create a “discrete gamma model”, which uses several categories to approximate the continuous gamma distribution. Here rates over sites are regarded as random variables drawn from a discrete distribution. This leads to more efficient estimation, easier interpretation of results, and also better fit of the model to data than models with constant rates. Hence we use it to measure the rate heterogeneity of the real data in the following chapters.

In this chapter, the various substitution models of the nucleotide and amino acid and the rate heterogeneity are described with detail respectively in section 2.1 and 2.2. Then the procedure of calculating likelihoods is demonstrated by an example with three taxa.
2.1 Substitution Model

We deal with two kinds of data in this thesis: nucleotide and amino acid. Their substitution models are discussed in the following respectively.

2.1.1 DNA Sequences

Four kinds of nucleotides, a, c, g and t, constitute DNA sequences. The mathematical expression of a substitution model is a table of rates (substitutions per site per unit evolutionary distance) at which each nucleotide is replaced by each alternative nucleotide. These rates can be expressed as a $4 \times 4$ instantaneous rate matrix, $Q$, in which each element $Q_{ij}$ represents the rate of change from base $i$ to base $j$ during some infinitesimal time period $dt$. The probability of a change from state $i$ to state $j$ at a given site does not depend on the history of the site prior to its being in state $i$. For example, if a DNA sequence position has base $a$ at some time $t_0$, the probability that it will have base $t$ at a later time $t_1$ depends only on the fact that it has base $a$ at $t_0$; knowing that it had state $c$ at some time prior to $t_0$ would be irrelevant to the probability.

The general time-reversible (GTR) model (Yang 1994) is a popular model. Almost all the other time reversible models proposed in the literature are its special forms. It is represented by

$$Q = \mu \ast \begin{bmatrix}
\cdot & a\pi_c & b\pi_g & c\pi_t \\
a\pi_a & \cdot & d\pi_g & e\pi_t \\
b\pi_a & d\pi_c & \cdot & f\pi_t \\
c\pi_a & e\pi_c & f\pi_t & \cdot 
\end{bmatrix}$$

where the rows and columns correspond to the bases $a$, $c$, $g$ and $t$, and $Q_{ii}$ ("." in the matrix) is defined as $-\sum_{j\neq i} Q_{ij}$. The diagonal elements in matrix $Q$ satisfy the condition that the sum of each row equals to 0. $\pi_a$, $\pi_c$, $\pi_g$ and $\pi_t$ are the equilibrium frequencies and the reversibility condition holds, i.e., $\pi_i Q_{ij} = \pi_j Q_{ji}$. The factor $\mu$ represents the mean instantaneous substitution rate. The average rate of substitution at equilibrium is 1, i.e. $-\sum_i \pi_i Q_{ii} = 1$.

If $a = c = d = f = 1$ and $b = e = \kappa$, the GTR model reduces to the HKY model
(Hasegawa et al. 1985), which is represented by

\[
Q = \mu \ast \begin{bmatrix}
\pi_c & \kappa \pi_g & \pi_t \\
\pi_a & \pi_g & \kappa \pi_t \\
\kappa \pi_a & \pi_c & \pi_t \\
\pi_a & \kappa \pi_c & \pi_g \\
\end{bmatrix}
\]

In HKY model, \( \kappa \) is the only specified parameter besides the base frequencies, presenting the transition bias.

If \( a = c = d = f = 1 \), \( b = 1 + K/\pi_R \) and \( e = 1 + K/\pi_Y \), where \( \pi_R = \pi_a + \pi_g \) and \( \pi_Y = \pi_c + \pi_t \), the GTR model reduces to the F84 model (Felsenstein and Churchill 1996), which is represented by

\[
Q = \mu \ast \begin{bmatrix}
\pi_c & (1 + K/\pi_R)\pi_g & \pi_t \\
\pi_a & \pi_g & (1 + K/\pi_Y)\pi_t \\
(1 + K/\pi_R)\pi_a & \pi_c & \pi_t \\
\pi_a & (1 + K/\pi_Y)\pi_c & \pi_g \\
\end{bmatrix}
\]

where \( K \) is the parameter determining the transition/transversion ratio.

But in order to calculate likelihoods we need the probabilities of change from any nucleotide to any other along a branch of length \( t \). Branch lengths in a tree, or time \( t \), are defined as the expected number of substitutions per site accumulated during that time period. There are 16 possible changes from one nucleotide to another (including remaining unchanged), and we organize the probabilities of those changes in a \( 4 \times 4 \) matrix, called the substitution probability matrix, \( P \). The matrix below describes the probabilities with a branch length that is \( t \) times one substitution.

\[
P(t) = \begin{bmatrix}
P_{aa}(t) & P_{ac}(t) & P_{ag}(t) & P_{at}(t) \\
P_{ca}(t) & P_{cc}(t) & P_{cg}(t) & P_{ct}(t) \\
P_{ga}(t) & P_{gc}(t) & P_{gg}(t) & P_{gt}(t) \\
P_{ta}(t) & P_{tc}(t) & P_{tg}(t) & P_{tt}(t) \\
\end{bmatrix}
\]

where \( P_{ac}(t) \) is the probability that a site starts with nucleotide \( a \) and ends with nucleotide \( c \) after the time interval \( t \), and so on. The diagonal elements \( P_{aa}(t) \), \( P_{cc}(t) \), \( P_{gg}(t) \) and \( P_{tt}(t) \) are the probabilities that a site starts and ends with the same
nucleotide type during $t$. The rows sum to 1, but the columns don't sum to anything in particular. It can be calculated as

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

The exponential of a matrix can be evaluated by decomposing the matrix into its eigenvalues and eigenvectors. Suppose the matrix $Q$ can be expressed by $HDH^{-1}$, where $H$ is the matrix of eigenvectors and $D$ the diagonal matrix of the eigenvalues of $Q$.

$$Q^i = (HDH^{-1})^i = HD^iH^{-1}$$

$$\implies e^Q = \sum_{i=0}^{\infty} \frac{1}{i!}Q^i = \frac{1}{i!} \sum_{i=0}^{\infty} HD^iH^{-1}$$

$$= H \left[ \sum_{i=0}^{\infty} \frac{1}{i!}D^i \right] H^{-1} = He^DH^{-1}$$

$$\implies e^{Qt} = He^{Dt}H^{-1}$$

Note that $e^D$ denotes the diagonal matrix with $e^{D_{ii}}$ as the diagonal elements, where $D_{ii}$ is the $i$th diagonal element in the diagonal matrix $D$.

2.1.2 Amino Acid Sequences

The genetic code can also be expressed as DNA codons which are composed of triplets of nucleotides. Each codon encodes for one of the 20 amino acids used in the synthesis of proteins with three exceptions where a termination codon is present. This produces some redundancy in the code: most of the amino acids being encoded by more than one codon. For simplicity, the 20 amino acids can be denoted by alphabets omitting B, J, O, U, X and Z. The relationships between codons, amino acids and the corresponding alphabets are listed in Table 2.1, in which the left row gives the first nucleotide of the codon, the top column gives the second nucleotide. The third nucleotides is given in the middle of the table along with the first two nucleotides. The corresponding amino acids with their three- and one-letter codes are also in the middle of the table.

Analogous substitution models can be defined for protein sequence data, except that there are 20 states rather than 4. It means that the substitution matrix of
<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>c</th>
<th>g</th>
<th>t</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>aca</td>
<td>gaa</td>
<td>aga</td>
<td>ata</td>
</tr>
<tr>
<td></td>
<td>act</td>
<td>thr</td>
<td>arg</td>
<td>ile</td>
</tr>
<tr>
<td></td>
<td>acc</td>
<td>asn</td>
<td>arg</td>
<td>met</td>
</tr>
<tr>
<td></td>
<td>acc</td>
<td>thr</td>
<td>ser</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>cac</td>
<td>gln</td>
<td>pro</td>
<td>cta</td>
</tr>
<tr>
<td></td>
<td>gcc</td>
<td>his</td>
<td>arg</td>
<td>val</td>
</tr>
<tr>
<td></td>
<td>cct</td>
<td>gln</td>
<td>arg</td>
<td></td>
</tr>
<tr>
<td>g</td>
<td>gaa</td>
<td>glu</td>
<td>ala</td>
<td>gta</td>
</tr>
<tr>
<td></td>
<td>gcc</td>
<td>glu</td>
<td>gly</td>
<td></td>
</tr>
<tr>
<td></td>
<td>gct</td>
<td>asp</td>
<td>val</td>
<td></td>
</tr>
<tr>
<td>t</td>
<td>taa</td>
<td>term</td>
<td>ser</td>
<td>tta</td>
</tr>
<tr>
<td></td>
<td>tca</td>
<td>term</td>
<td>trp</td>
<td>tgg</td>
</tr>
<tr>
<td></td>
<td>tcc</td>
<td>tyr</td>
<td>cys</td>
<td>tgc</td>
</tr>
<tr>
<td></td>
<td>tgg</td>
<td></td>
<td>trp</td>
<td>ttc</td>
</tr>
<tr>
<td></td>
<td>ttc</td>
<td></td>
<td></td>
<td>ttt</td>
</tr>
</tbody>
</table>

Table 2.1: The relationship between amino acids, three- and one-letter codes.

amino acids is much larger than DNA sequences. Estimating the matrix needs a
great many observations and consumes time. So instead of following the procedure
of the DNA substitution model, models of amino acids have concentrated on the
empirical approach. The two famous models for amino acid are the Dayhoff model
(Dayhoff et al. 1978) and the JTT model (Jones et al. 1992).

Originally, the protein evolution model is only given in the form of probability
matrices $P(t)$ relating to specific level of sequence divergence, in which $P_{ij}(t)$ gives the
probability that amino acid $i$ will change to amino acid $j$ at any single site after time
t $\geq 0$. Since there are 20 amino acids, $i$ and $j$ take the values 1,...,20. The probabilities
in $P(t)$ can estimated based on closely related pairs of sequences and by counting the
differences between them. Ideally every present-day sequence with its own immediate
predecessor should be compared and thus accurately map the evolutionary history
of each sequence position. Since it is impossible, the Dayhoff model (Dayhoff et al.
1978) used a “common ancestor” method to select pairs that were 85% identical on a
data set with 1572 changes in 71 groups and the JTT model (Jones et al. 1992) used
an approximate method (a pairwise present-day ancestor scheme) on 23,000 protein
sequences for inferring the phylogenetic relationships among the sequences in the data.
set. So generally speaking, the main difference between the Dayhoff and JTT models are the basic data sets and the methods to pick up the closely related pairs.

Let \( n_{ij} \) denote the observed number of occurrences of sites in all aligned sequences pairs with amino acids \( i \) in one sequence and \( j \) in the other. Because the direction of an evolutionary change cannot be determined from the observation, we make \( n_{ij} = n_{ji} \). For example, given the following alignment:

\[
\begin{align*}
\text{ADEGLF} \\
\text{ADECLA}
\end{align*}
\]

The following substitutions, \( G \rightarrow C, C \rightarrow G, F \rightarrow A, A \rightarrow F \), may have happened. The counts that the amino acid in column \( j \) is replaced by the amino acid in row \( i \) or vice versa decide \( n_{ij} \). If we assume the \( n_{ij} \) accurately represent all evolutionary events, an intuitive estimate for the rate of change of \( i \) to \( j \) \( (j \neq i) \) is given by the number of such events as a proportion of all observations of \( i \):

\[
Q_{ij} = \frac{n_{ij}}{\sum_k n_{ik}}
\]

This equation represents a simple way to estimate the substitution matrix \( Q \) directly from counts of observed differences between closely related sequences pairs. Note that in general it underestimates the actual numbers of changes because when multiple changes occur at a single site, they are observed as at most single replacements. However, in Dayhoff and JTT model, the \( Q \) matrix was not estimated directly but via corresponding probability matrices \( P(t) \). The detail descriptions are not given here, but can be obtained by checking the original paper (Dayhoff et al. 1978 and Jones et al. 1992).

Overall, the \( Q \) matrix here is analogous in form to the GTR model for nucleotide substitution. One important difference is that in contrast with the GTR nucleotide model, the values are fixed and not estimated for each new data set of interest.

### 2.2 Accommodating Rate Heterogeneity across Sites

In the early analyses, the model was assumed homogeneous across sites, meaning that the rate of substitution was the same at different sites. But this assumption is
unrealistic. In stochastic models, we can assume that the rate of evolution at each site follows some probability distribution.

Though discrete distributions have been used, it is believed that the distribution of rates is more likely to be continuous. The most common continuous distribution for modeling rate heterogeneity is the gamma (\( \Gamma \)) distribution (Yang 1993). The probability density function of the rate \( r_h \) of site \( h \) is

\[
f(r_h) = \beta^\alpha \Gamma(\alpha)^{-1} e^{-\beta r_h \alpha^{-1}}
\]

with mean \( E(r_h) = \alpha/\beta \) and variance \( V(r_h) = \alpha/\beta^2 \). To avoid the use of too many parameters, the mean of \( r_h \) is restricted to be 1, making \( \alpha = \beta \). So the \( \Gamma \) distribution depends on a single parameter \( \alpha \) that is equal to the inverse of the squared coefficient of variation of the substitution rate. A small \( \alpha \) suggests that rates differ significantly over sites, while a large value means almost equal rates. But the algorithm of obtaining likelihood using the \( \Gamma \) distribution involves intensive computation, making it impractical. Yang (1994) provides an alternative method called the "discrete gamma model" in which the \( \Gamma \) distribution is divided into several rate categories by finding boundaries in the distribution such that each category has equal probability. The mean of each category is used to represent all the rates within that category. This approximation fits the original model with as few as four rate categories and is more efficient in computation.

2.3 Likelihood

Suppose there are \( S \) species and each sequence has \( N \) sites. The two kinds of data can be represented as follows:

\[
\begin{align*}
\text{Species 1:} & \quad X_{11} \quad X_{12} \quad X_{13} \quad \ldots \quad X_{1N} \\
\text{Species 2:} & \quad X_{21} \quad X_{22} \quad X_{23} \quad \ldots \quad X_{2N} \\
\ldots & \quad \ldots \quad \ldots \quad \ldots \quad \ldots \\
\text{Species S:} & \quad X_{S1} \quad X_{S2} \quad X_{S3} \quad \ldots \quad X_{SN}
\end{align*}
\]

where \( X_{ij} \), \( i = 1, 2, \ldots, S \), \( j = 1, 2, \ldots, N \) is one of \( a, c, g \) and \( t \) in DNA sequences, or one of 20 amino acids. The whole data is written as \( X \) and the data of \( h \)th site \((X_{1h}, \ldots, X_{Sh})^T \) is written as \( X_h \).
We begin by considering an informal explanation about why we introduce likelihood into the tree topology problem. If we only have data for one site of $S$ species, we can say that the two species that have the same states are more similar than the two species that have different states. But now we have data for many sites. Suppose at the first site, the first two species have the same states but the third species doesn't; at the second site, the first and third species have the same states but the second doesn't. We can't easily decide how to represent their true relationship in a $S$-taxon tree. The likelihood function plays a role to solve this problem because it is the vehicle that gives the joint probability of the data for particular parameters.

For a given unrooted tree (in Figure 2.1), $X = \{x_{1h}, x_{2h}, x_{3h}\}$ (terminal nodes) stands for the states in the $h$th column of the data set. Let $y_h$ be the state of the interior node of the tree, which is one of the possible states. $\pi_{y_h}$ is the probability of state of $y_h$ appearing. The branch lengths in the tree are $t_1$, $t_2$ and $t_3$.

![Figure 2.1: Tree 1](image)

We don't know what the state of $y_h$ is. It may well be the same as the state of $x_{1h}$, $x_{2h}$ or $x_{3h}$, but it could be anything. We need to sum over all the possibilities for node $y_h$. The likelihood of rooted and unrooted trees are the same. This means that wherever the root is, the likelihoods calculated from a single tree topology are the same. Here $x_{1h}$ is used as the root in the following computation. Treating the observations with
homogeneous rate, the likelihood of the $h$th site is

$$f(X_h|\theta) = \sum_{y_h} \pi_{x_{1h}y_h} P_{x_{1h}y_h}(t_1)P_{y_hx_{2h}}(t_2)P_{y_hx_{3h}}(t_3)$$

where $P_{x_{1h}y_h}(t_1)$ is the substitution probability from the nucleotide of $x_{1h}$ to that of $y_h$ during the time $t_1$ and $\theta = \{t_1, t_2, t_3\}$. Here $\theta$ denotes the parameters during calculation, such as the branch lengths $t_i$, $i = 1, 2, 3$ and the substitution model. If the heterogeneous rate exists (such as with distribution of "discrete gamma"), the likelihood of each site should be calculated by a weighted sum over rate factors.

Under the assumption that the sites are independent, each topology $\tau_i$ has likelihood

$$L(X|\theta_i, \tau_i) = \prod_{h=1}^{N} f(X_h|\theta_i, \tau_i)$$

Because the likelihood is for the entire sequence, its value is very small when $N$ is large. It is better to use log-likelihood, defined as

$$l(X|\theta_i, \tau_i) = \log L(X|\theta_i, \tau_i)$$

It is unrealistic to get the true value of $\theta$ to calculate likelihood. The typical treatment is to use $\hat{\theta}$ which maximizes the likelihood as the estimate of $\theta$. $\hat{\theta}_i = \arg\max_{\theta_i} \{l(X|\theta_i, \tau_i), \theta_i \in \Theta\}$ is the maximum likelihood estimate of $\theta_i$ given a fixed tree topology $\tau_i$. The profile maximum likelihood for a given tree $\tau_i$ is

$$L_i(\theta) = L(X|\hat{\theta}_i, \tau_i)$$

and the profile maximum log-likelihood is

$$l_i = l(X|\hat{\theta}_i, \tau_i) = \log L_i(\theta)$$

The tree with maximum likelihood, $l_{ML} = \max\{l_1, l_2, ..., l_m\}$, is called the maximum likelihood (ML) tree, where $m$ is the total number of possible trees. It is considered to be the best estimate of the true tree.

As the essential element, likelihood will be mentioned frequently in the following chapters. The details of calculating likelihood won’t be given any more.
Chapter 3

The Comparison of the Confidence Regions

In phylogenetic analysis, an important issue is to determine the confidence regions for gene trees from sequence data. A number of the existing confidence region methods are likelihood based and look for trees that are close to the maximum likelihood (ML) tree in some sense. Interestingly, many available procedures can give rather different confidence regions (e.g. Goldman, Anderson and Rodrigo 2000; Strimmer and Rambaut 2001; Shimodaira 2002).

In this chapter our main aim is to evaluate the performance of a number of the current proposals and modifications of them to see which procedure gives us the best result under different conditions. Our approach will be to construct the confidence regions based on tests of the null hypothesis that the tested tree has the same topology as the true one. For simplicity, we will refer to $\tau_0$ as the tree topology being tested. The branch lengths are not specified in the null hypothesis but are estimated during the test. In most tests the ML branch lengths which maximize the likelihood of the given tree structure are used. The $P$-value is calculated as the probability, under $H_0$, of observing a value of the test statistic at least as large as that observed. Using the general duality between testing and confidence regions, the test methods can be used to construct confidence regions of topologies. A $100(1-\alpha)\%$ confidence region for the true topology is a set of topologies that contains the true topology with probability $1-\alpha$. It includes $\tau_0$ if the null hypothesis is not rejected with significance level $\alpha$. The probability that the confidence region includes the true tree is called the coverage.

We will examine the following test procedures: Shimodaira-Hasegawa (SH) test (Shimodaira and Hasegawa 1999), single distribution nonparametric bootstrap (SDNB) test (introduced in this chapter), SOWH test (Swofford et al. 1996; Hillis, Mable and Moritz 1996; Goldman, Anderson and Rodrigo 2000), star form of SOWH (SSOWH)
test (Antenaza 2003), single distribution parametric bootstrap (SDPB) test (introduced in this chapter), approximately unbiased (AU) test (Shimodaira 2002), likelihood weight (LW) test (Strimmer and Rambaut 2001) and generalized least squares (GLS) test (Susko 2003). Except for the GLS test which is distance based, all the other tests are based on likelihood. The simulation results are compared under two criteria: the coverage and the size, which is defined here as the number of trees in the confidence regions.

The rest of this chapter is organized as below. Section 3.1 introduces the statistical methods including the detailed description of the two newly proposed SDNB and SDPB tests and their relationships with an emphasis on methodology. The comparison of confidence regions of gene trees according to their coverages and sizes based on data sets simulated from trees with small and large numbers of taxa is presented in section 3.2 with the relevant discussion in section 3.3. Without considering model misspecification, the SOWH test performs very well with reasonable coverage and small size of the confidence region. The SDNB and SDPB tests based on two different bootstrap methods also have small sizes, but undercover the true tree (the coverage of the true tree is lower than the nominal coverage) in some cases. The SH, LW, AU and SSOWH tests always give conservative results in the simulations, i.e. the coverage is larger than the nominal coverage, giving large confidence regions. Although under large sample approximation, the coverages of the AU and GLS tests are expected to be the same as the confidence level, it didn’t happen in the simulations. Their asymptotic characters are not approached by the sequence lengths in the simulation and it is hard to tell how long the sequence should be to guarantee the asymptotic characters. Nonparametric and parametric bootstraps behave differently, which is demonstrated by looking at the critical values for the SDNB and SDPB tests. Finally the SDNB test with the AU test as an alternative is recommended for the real data in section 3.4.

3.1 Methodology

In phylogenetic analysis, an important issue is to determine the confidence regions for gene trees from DNA sequences. In carrying out the tests to construct the confidence
regions, the overall biological goal is the same: are the observations compatible with a specified tree topology $\tau_0$? This can be written as

$$H_0: \text{true topology } = \tau_0$$

which is called the canonical hypothesis. $\delta_i = l_{ML} - l_i$ is commonly used as the test statistic of the canonical hypothesis, where $l_i$ denotes the log-likelihood of tree $i$ and $l_{ML}$ denotes the maximum log-likelihood of trees under consideration.

The bootstrap method is widely used for constructing the distribution of the test statistic when a closed form expression is not available. There are two types of bootstrap: nonparametric and parametric. For DNA sequences, the combinations of nucleotide types in sites are considered as the samples in nonparametric bootstrap. The sites are resampled with replacement to build the replicate data sets. In the standard bootstrap, all the nuisance parameters, such as the branch lengths and the base frequencies, need to be reestimated to calculate the likelihoods. The RELL technique gives time-saving approximations, in which the site likelihoods are resampled. It is equivalent to calculating the likelihoods of the standard replicates with the parameters estimated from the original data set. These two techniques give similar results with the latter being more popular. In the parametric bootstrap, the replicate data sets are simulated from a parametric substitution model and some tree. For instance, with the SSOWH test, bootstrap data is simulated from the star topology with estimated substitution model parameters and external branch lengths while for the SOWH test, bootstrap data is simulated from the hypothesized tree with estimated branch lengths. Hence the parametric bootstrap relies on the model specification more closely than the nonparametric bootstrap, which also makes it more powerful (Sokal and Rohlf 1995). However this also is a limitation due to the concern that some assumptions in the model may not be realistic. Any symbol with * in this chapter refers to estimates obtained from a replicate data set and will be referred to as a bootstrap value. For instance, $l_{ML}$ is the log-likelihood of the ML tree of the original data set and $l'_{ML}$ would denote the corresponding value of a bootstrapped data set.

We now describe the tests in terms of the hypothesis, test statistic and the bootstrap method. The names of the tests and the corresponding acronyms are listed in
Table 3.1.

<table>
<thead>
<tr>
<th>Tests</th>
<th>Acronyms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shimodaira-Hasegawa test</td>
<td>SH</td>
</tr>
<tr>
<td>Nonparametric Bootstrap test</td>
<td>SDNB</td>
</tr>
<tr>
<td>SOWH test</td>
<td>SOWH</td>
</tr>
<tr>
<td>Star version of SOWH test</td>
<td>SSOWH</td>
</tr>
<tr>
<td>Parametric Bootstrap test</td>
<td>SDPB</td>
</tr>
<tr>
<td>Approximately Unbiased test</td>
<td>AU</td>
</tr>
<tr>
<td>Likelihood Weight test</td>
<td>LW</td>
</tr>
<tr>
<td>Generalized least squares test</td>
<td>GLS</td>
</tr>
</tbody>
</table>

Table 3.1: The tests and their acronyms.

3.1.1 The SH Test

The SH test (Shimodaira and Hasegawa 1999) is a modified version of the KH test (Kishino and Hasegawa 1989). The KH test was designed to test any two topologies selected independently of any analysis of the data, but was often inappropriately used for comparing a candidate tree with the ML tree, a fact noted by many authors (Shimodaira and Hasegawa 1999; Goldman, Anderson and Rodrigo 2000). We won’t give details and simulation results for the KH test since we are interested in more than two candidate trees. The hypothesis for the SH test is

\[ H_0: \text{The observations are explained equally well by } \tau_1, \tau_2, ..., \tau_m \]

where \( m \) is the number of candidate trees. The hypothesis indicates that we are testing a group of trees rather than trees independently. Though the hypothesis is not canonical, the canonical test statistics are used in this test,

\[ \delta_i = l_{ML} - l_i, \quad i = 1, 2, ..., m \]

and their distributions are estimated by nonparametric bootstrap. The process of bootstrap is as follows.

1. Generate \( B \) nonparametric bootstrap replicates. In the replicate data set, calculate the maximum likelihoods over parameters \( \hat{\theta}_i^* \) for each permitted topology. The optimal log-likelihood values \( l_i^* \) are stored in a \( m \times B \) matrix with the rows for the candidate trees and the column for the replications.
2. Subtract the average of each row from the entries of the array. The data of the matrix \( \tilde{l}^*_i \) are replaced by \( \tilde{l}^*_i = l^*_i - \bar{\tilde{l}}^*_i \), \( i = 1, \ldots, m \), where \( \bar{\tilde{l}}^*_i \) denotes the mean of \( B \) replicate log-likelihood of \( \tau_i \). This step is referred to as "centering".

3. For each column of the array, find \( \tilde{l}_{ML}^* \), the maximum over \( \tau_i \) of the adjusted log-likelihoods \( \tilde{l}^*_i \) and replace the matrix by bootstrap replicate statistics \( \delta^*_i = \tilde{l}_{ML}^* - \tilde{l}^*_i \). \( \delta^*_i \) is a replicate of \( \delta_i \).

4. Test whether the attained values of \( \delta_i \)'s, \( i = 1, 2, \ldots, m \) are plausible samples from \( \delta_i^* \). For significance level \( \alpha \), if \( \delta_i \) is less than the \( 100(1-\alpha) \)th percentile of \( \delta_i^* \), \( \tau_i \) is included in \( 100(1 - \alpha)\% \) confidence region.

Here we give an intuitive explanation and some comments on the rationale and the effects of the "centering" procedure proposed in the resampling method of the SH test. The nonparametric bootstrap approximates the true distribution whether the null hypothesis is true or not. Since the null hypothesis may not be true, it may not be approximating the distribution of the test statistic under the null hypothesis as desired. If all the trees equally support the observations, they should have equal expected log-likelihoods, denoted by \( E[l_i], i = 1, \ldots, m \). In order to build the distribution of the test statistic under \( H_0 \), the least favorable configuration (l.f.c.), i.e. \( E[l_1] = E[l_2] = \ldots = E[l_m] \), is considered. The l.f.c. states zero branch length for any split that is not present in all the trees under consideration. It is least favorable in the sense that it corresponds to the specific hypothesis consistent with \( H_0 \) which is the most difficult to reject. By subtracting and adding \( E[l_1] \) to \( \delta_i \), we have

\[
\delta_i = l_{ML} - l_i = \max(l_1, \ldots, l_m) - l_i = \max(l_i - E[l_1], \ldots, l_m - E[l_1]) - (l_i - E[l_1])
\]

Under the l.f.c assumption, \( E[l_1] \) can be replaced by \( E[l_i] \) for all \( i \), so that

\[
\delta_i = \max(l_i - E[l_1], \ldots, l_m - E[l_m]) - (l_i - E[l_i])
\]

Let \( \bar{\tilde{l}}^*_i, i = 1, \ldots, m \) denote the mean of all replicate log-likelihoods of \( \tau_i \). The "centering" procedure replaces \( E[l_i] \) by \( \bar{\tilde{l}}^*_i \) to calculate the bootstrap value of \( \delta_i \) in (4), i.e.

\[
\delta_i^* = \max(l_i^* - \bar{\tilde{l}}^*_1, \ldots, l_i^* - \bar{\tilde{l}}^*_m) - (l_i^* - \bar{\tilde{l}}^*_i)
\]
However, since the l.f.c. assumption might not necessarily hold for \( \tilde{l}_1, \ldots, \tilde{l}_m \), the equivalence between (3.3) and (3.4) does not hold for the bootstrap value of \( \delta^*_i \). In fact, we always get larger bootstrap values of \( \delta_i \) when more trees are involved in the null hypothesis, which results in larger \( P \)-values making it harder to reject the tested trees.

### 3.1.2 The SDNB Test

The bootstrap paradigm requires us to replace the true parameters with the estimates and the estimates with bootstrap values in replicates. In phylogeny, the tested tree topology is the parameter we are interested in. As the estimate of the true tree, the ML tree obtained from the original data set gives the most probable evolution process. So instead of "centering", we replace the tested tree by the ML tree of the original data set in bootstrap replicates in term of bootstrap theory. This is the core idea of our proposed method. And the proposed method, single distribution nonparametric bootstrap (SDNB) test, only requires a single distribution for all the test statistics.

The hypothesis and the test statistic are the canonical ones in the SDNB test.

The test statistics are

\[
\delta_1 = l_{ML} - l_1, \ldots, \delta_m = l_{ML} - l_m
\]

and the ML tree, \( \tau_{ML} \), is recorded.

The key steps of constructing the distribution of the test statistics are:

1. Generate \( B \) nonparametric bootstrap replicate data sets.
2. Reestimate any free parameters to get maximum log-likelihood \( l_{ML}^* \) and \( l_{ML} \) for each replicate. \( l^* \) denotes the log-likelihood for the bootstrap data set. The subscripts ML, denotes the ML tree of the original data set and ML* denotes the ML tree of the replicate.
3. Construct the distribution of \( \delta_i = l_{ML} - l_i \) by the difference of log-likelihood between the ML tree of the replicate and the ML tree of the original data set, \( \delta^* = l_{ML}^* - l_{ML}^* \). Comparing the calculation of \( \delta^* \) in the bootstrap to the original test statistic, the tested tree \( \tau_i \) is replaced with \( \tau_{ML} \) and \( \tau_{ML} \) with \( \tau_{ML}^* \). (We can use RELL in this step to save time. In this chapter the test was implemented without RELL.)
4. Test whether the attained values of \( \delta_i \)'s, \( i = 1, 2, \ldots, m \) are plausible samples from
$\delta^*$. For significance level $\alpha$, if $\delta_i$ is less than the $100(1-\alpha)$th percentile of $\delta^*$, $\tau_i$ is included in $100(1-\alpha)$% confidence region.

3.1.3 The SOWH and SSOWH Tests

The SOWH test was named after the authors who originally described it (Swofford et al. 1996; Hillis, Mable and Moritz 1996). As with the SDNB test it has the canonical hypothesis and test statistic. In this test, a parametric bootstrap is used to construct the distribution of the test statistic. The replicate data sets are simulated under the tested tree with the nuisance parameters, such as the branch lengths and the substitution model, estimated from the original observations. The distribution of $\delta_i$ has to be built for each tested tree $\tau_i$ using $\delta_i^* = l_{ML}^* - l_i^*$, which is computationally intensive. The process of parametric bootstrap is as follows.

1. Generate $B$ parametric bootstrap replicates. Using the given tree topology $\tau_i$ and the ML estimates of $\theta_i$ from the original data, for example the branch length and substitution model, to simulate replicated data sets.

2. Reestimate any free parameters to get maximized log-likelihoods $l_j^*$ of $\tau_j$, $j = 1, \ldots, m$ and record the maximum log-likelihood $l_{ML}^*$ and the log-likelihood of $\tau_i$, $l_i^*$.

3. Calculate $\delta_i^* = l_{ML}^* - l_i^*$. The data set of $\delta_i^*$ gives an estimate of the distribution of $\delta_i$ under $H_0$.

4. Test whether the attained value of $\delta_i$ (from the original data) is a plausible sample from the distribution of $\delta_i^*$. $\tau_i$ is included in $100(1-\alpha)$% confidence region if $\delta_i$ is less than $100(1-\alpha)$th percentile of $\delta_i^*$.

Treating $H_0$ as the single null hypothesis, there are two disadvantages to this test. One is low coverage when the true tree topology is close to a star topology. (Detailed description will be given in the last paragraph of Section 3.3.1) The other is that the distribution of $\delta_i$ in the SOWH test has to be built for each tested tree $\tau_i$ using $\delta_i^* = l_{ML}^* - l_i^*$, which is computationally intensive.

In order to solve the first problem, Antezana (2003) proposed a star version of parametric bootstrap by which the replicates are simulated from the star topology with ML branch lengths estimated from the original observations. This is similar to the treatment of the complex null hypothesis to calculate the largest $P$-value under
$H_0$. It is motivated by the fact that the star topology can be viewed as the boundary between trees. Using the boundary tree of $H_0$ to simulate data will naturally make the tested tree difficult to reject. Since the tree model used to generate the parametric bootstrap is the only difference between these two tests, the later one is referred to as the star version of the SOWH (SSOWH) test. Its process is similar to the SOWH test except that step 1 is replaced by $1'$. Generate $B$ parametric bootstrap replicates. The replicates are simulated from a star topology with the ML estimate of the external branch lengths (there are no internal branches in a star topology) and substitution model.

The underlying assumption of this star version of parametric bootstrap is that all the tree topologies equally support the observations, which is similar to the assumption of the null hypothesis in the SH test. So the two methods work well when the true tree is close to a star topology.

### 3.1.4 The SDPB Test

Following the bootstrap paradigm of the SDNB test, computational intensity is decreased in the single distribution parametric bootstrap (SDPB) in which the replicates are simulated from the ML tree of the original data set. The SDPB test can be considered as the parametric bootstrap version of the SDNB test. The key steps of this method are the same as the SDNB test except step 1, which is replaced by:

1'. Generate $B$ parametric bootstrap replicates based on $\tau_{ML}$ with the ML branch length and other parameters obtained from the original observations.

This procedure gives the SDPB test a substantial computational advantage over the SOWH test since $\delta_i$'s are all evaluated using the distribution of $\delta^* = l_{ML}^* - l_{ML}^\star$. This eliminates the need of a separate bootstrap for each $\tau_i$ which is required in the SOWH test. The SDPB test is much more feasible than the SOWH test, especially for trees with large numbers of taxa and consequently large number of trees to test.

### 3.1.5 Other Methods

The AU (almost unbiased) test (Shimodaira 2002) is widely used now. As we know, the log-likelihood of $\tau_i$ is the sum of log-likelihoods of all sites. Shimodaira considered
$l_i$ as a random variable with the expected value as $\mu_i$. As the sequence length $N$ goes to infinity, we expect that the likelihood of the true tree is always larger than the other trees. That's why the expectations of the log-likelihoods are compared in the null hypothesis of AU. Therefore the hypothesis for testing $\tau_i$ is

$$H_0 : \mu_i \geq \mu_j, \quad j = 1, \ldots, m$$

The proportion of times that $\tau_i$ is the ML tree in the bootstrap replicates is called bootstrap probability (BP). A multiscaled nonparametric bootstrap in which different sequence lengths are used provides a technique for computing BP. Since it gives the first order accuracy for the $P$-value (Efron, Halloran and Holmes 1996), Shimodaira used the results from Efron and Tibshirani (1998) to correct BP to give the second order accurate $P$-values which means the expected coverage of the AU test is $(1 - \alpha) + o(n^{-3/2})$ (see equation 11 and the discussion therein of Shimodaira, 2002).

The key idea to find the BP values of different sequence lengths and calculate $P$-values is as follows.

1. Specify the scaling constants $r_1, \ldots, r_G$ and generate $B_g$ nonparametric bootstrap data sets with different scaled sequence lengths $n'_g = r_g \cdot n$, $l = 1, \ldots, G$. The number of bootstraps $B_g$ may be same or different for different sequence lengths.

2. Count the proportions of times $BP_g$ that $\tau_i$ is the ML tree in the bootstrap replicates with sequence lengths of $n'_g$.

3. Estimate $d$ and $c$ by the weighted least squares method to minimize the residual sum of squares

$$RSS(d, c) = \sum_{g=1}^{L} v_g^{-1} \{d \sqrt{r_g} + c \sqrt{r_g} - \Phi^{-1}[1 - BP_g]\}^2,$$

where $v_g$ is the variance given by

$$v_g = BP_g[1 - BP_g]/(\phi(\Phi^{-1}(BP_g))^2 B_g$$

Note that $\Phi(.)$ and $\phi(.)$ are the quantile function and the density function of the standard normal distribution.

4. Calculate the $P$-value as $1 - \Phi(d - c)$.
The LW (likelihood weights) test is designed to construct confidence regions directly and does not involve any tests of hypotheses (Strimmer and Rambaut 2001). Likelihood weight is the proportion of the likelihood of a tested tree over the sum of the likelihoods of all candidate trees.

$$LW_i = \frac{L_i}{\sum_{j=1}^{m} L_j}$$ (3.6)

Nonparametric bootstrap gives the estimate of the expected likelihood weight of each tree. Tested trees are ordered by expected likelihood weights and will be included in the confidence region until the sum of their weights is equal to or slightly greater than the confidence level. Trees with larger expected likelihood are more likely to be the true tree and hence will be in the confidence region.

The process of this method is given below.

1. Generate $B$ nonparametric bootstrap data sets.
2. Calculate the likelihoods of all the candidate trees for each replicate and store them in a $m \times B$ matrix with the rows for the candidate trees and the column for the replicates.
3. Compute the proportion of the likelihood over the sum of all the likelihoods in the corresponding column and average the proportions over $B$ replicates to obtain the likelihood weights. The sum of the likelihood weights is 1.
4. Rank the likelihood weights with descending order and include the tree $\tau_i$ in 100(1 – $\alpha$)% confidence region if the sum of the likelihood weights larger than it is less than 1 – $\alpha$.

The GLS test (Susko 2003) is based on the idea of fitting ML pairwise distances using a regression model determined by the tree topology. It is computationally inexpensive relative to the ML methods, especially for trees with large number of taxa. All candidate trees need to be tested separately since the canonical hypothesis is used. The ML pairwise distances, consistent estimates of the true distance, form the dependent variable vector $Y$. They can also be expressed as the sum of the branch lengths based on the tree topology. Therefore, the tested tree topology is transformed into a design matrix $X$ given the arrangement of the ML pairwise distances and the branch lengths in the vectors. For a pair of taxa $i$ and $j$, the index of the branch is 1.
in the design matrix if the path from $i$ to $j$ passes through it; 0 otherwise. The branch lengths $T$ are estimated to minimize $(Y - XT)^T V^{-1} (Y - XT)$ by the least square method, where $V$ is the covariance matrix for the estimated ML pairwise distance. Since the ML pairwise distances are asymptotically multivariate normal distributed, $(Y - XT)^T V^{-1} (Y - XT)$ asymptotically follows a chi-squares distribution with degrees of freedom $S(S - 1)/2 - (2S - 3)$, where $S$ is the number of taxa. (This method is demonstrated on an example with 4 taxa in section 4.2.)

3.2 Simulation Methods and Analysis

3.2.1 Trees

For the comparison, we used data simulated from model trees using Seq-Gen, version 1.2.5 (Rambaut and Grassly 1997) so that the true tree topologies are known. The model trees with small and large number of taxa were considered separately.

Small Number of Taxa

5-taxon trees with five external branches and two internal branches were considered as models for simulation. In the model trees, two external branches were set to be the long branches with the same length, the others were the short branches again with the same length. So there are three possible tree shapes determined by the positions of the two long branches as shown in Figure 3.1. The values of the factors affecting the results of phylogenetic analyses, such as sequence lengths, branch lengths and tree shapes, are listed in the following table.
sequence length  500  1000
long branch length  0.9  1.5
short branch length  0.03  0.06
tree shape (in Figure 3.1)

Figure 3.1: The three tree shapes with a,...,g denoting the branch lengths.

There are 12 different model trees labeled A to L as listed in Table 3.2. The branch lengths in each tree are given following the order of a,...,g as in Figure 3.1. Trees A to D correspond to the tree on the left in Figure 3.1, E to H are for the tree in the middle and I to L for the tree on the right. The percentage of time that the ML and the true tree (PMLT) is same is also listed in the table for different sequence lengths. The smaller PMLT is, the more difficult the phylogenetic analysis is. The PMLT is highly related to the coverages of the tests, since the ML tree is always in the confidence region. It is useless to compare the coverages of the confidence regions if all the PMLTs are close or higher than the confidence level. The last column shows the diameters of the trees, which are the maximum distance between any two taxa in a tree (Zwickl and Hillis, 2002).

Large Number of Taxa

The simulations were also performed for trees with large number of taxa. In order to make the work consistent with reality, one data set was simulated from a published
Table 3.2: The branch lengths, the PMLT's of different sequence length (sl) and the diameters of the model trees.

66-taxon tree (Murphy et al. 2001) with sequence length of 3000. We selected three subsets of taxa of size 10, 15 and 20 from the 66 taxa and constructed the ML tree for each by PAUP*. The three ML trees serve as the model trees for large number of taxa. The tree topologies are shown in Figure 3.2 and the detailed tree structures including branch lengths are given in the appendix.

3.2.2 Substitution Models

In order to generate data sets by Seq-Gen, we must specify the substitution models. A GTR model is used to generate data sets for small number of taxa with base frequencies as \((\pi_a, \pi_c, \pi_g, \pi_t) = (0.18, 0.33, 0.26, 0.23)\) and \((a/f, b/f, c/f, d/f, e/f) = (3.297, 12.55, 1.167, 2.06, 13.01)\). It requires one parameter less to determine the substitution model when \(a, b, c, d\) and \(e\) are divided by \(f\), which makes \(f/f\) always equals to 1. For likelihood based approaches it becomes a heavy computational burden to find the ML tree for large number of taxa. In order to save time, we use HKY model with base frequencies as \((\pi_a, \pi_c, \pi_g, \pi_t) = (0.37, 0.24, 0.12, 0.27)\) and transition/transversion ratio \((\kappa)\) as 2.93. These values were taken from the paper by Zwickl and Hillis (2002), which are estimated by maximum likelihood on a tree obtained by a parsimony search for two of the genes present in the Murphy et al. (2001) data set (12S rRNA and 16S rRNA).
Figure 3.2: The structures of 10-, 15- and 20-taxon trees for simulation

1, a protein-coding gene). To avoid confounding issues, we didn’t consider heterogeneous site rates. Since we didn’t consider model misspecification, the models for simulation and analysis were consistent.

3.2.3 Methods

All the likelihoods were calculated from PAUP* 4.0b8 (Swofford 2000). The SH test was implemented directly by the programs in PAUP* with 1000 bootstraps of RELL. After the sitewise log-likelihoods were obtained from PAUP*, the AU test was performed using CONSEL (Shimodaira and Hasegawa 2001) which also uses RELL for bootstrapping. We used the default settings of CONSEL with the
scales in the bootstrap from 0.5 to 1.4, in increments of 0.1 and with 10000 bootstrap
straps for each scale. The expected likelihood weights were estimated by the mean
of 1000 replicates in the LW test. The distribution of the test statistics in the
SOWH, SSOWH, SDNB and SDPB tests were built based on 100 bootstrap replicates.
The GLS test used a program developed from glsDNA_eig written by Susko.
When the eigenvalues of the estimated covariance matrix are greater than $10^{-6}$, this
program performs the test discussed in Susko (2003). Otherwise a modification is
automatically implemented to avoid difficulties with the near-singularity of the co-
variance matrix (more details are given in the software documentation available at

All the possible 15 unrooted tree topologies of 5 taxa were tested by all methods to
determine whether or not they belong to the confidence region. When the number of
taxa becomes large, it is difficult to construct confidence region based on all possible
trees. Generally we are interested in the trees with highest likelihoods and the ML
tree. We tested the 100 trees with the highest likelihood by a search method, such
as nearest-neighbor-interchanging (NNI) branch swapping, to get a measure of the
size of the confidence region. To evaluate coverage, we only need to check whether
the true tree is in the confidence region. It should be noted that under a searching
method the "ML" tree we obtained may not be the global ML tree.

The 100(1-α)% confidence region was obtained by collecting the trees with $P$-value
greater than significance level $\alpha$ in tests. Note that the LW test uses likelihood weights
rather than $P$-values to construct the confidence region. If we construct confidence
regions only based on 100 repeated simulations for each method, the variance of
the coverage should be considered. An approximate 95% confidence interval for the
nominal coverage $c$ can be estimated as

$$(c - 1.645\sqrt{\frac{c(1-c)}{100}}, 1)$$

We considered one-sided confidence interval since only low coverage causes problems.
When $c$ is 0.95 or 0.8, the corresponding 95% confidence interval is (0.91, 1) or (0.76,
1).
3.2.4 Simulation Results

Small Number of Taxa

We simulated 100 data sets from each model tree in Table 3.2 with sequence lengths of 500 and 1000. Confidence regions were constructed for all of them based on eight methods. The estimated coverage is the proportion of times that the true tree is in the confidence region during the simulations. Note that it will always exceed the PMLT for all procedures except the GLS test. Our goal is to have a small size of confidence region with coverage at least $1 - \alpha$. The size is simply the number of trees (out of 15) in the confidence region. From a testing point of view, including a tree other than the true tree in the confidence region is equivalent to making a type-II error. The mean size of 100 confidence regions under the same simulation parameters is a good indication of the size of the confidence region.

Through the simulation results we found that the behavior of the methods was consistent regardless of the different simulation parameters. The mean sizes of the confidence regions are almost always in the ascending order according to: SDPB, SOWH, SDNB, SSOWH, GLS, LW, AU and SH. (The mean sizes of the SOWH and SDNB tests are close. It is likely that the latter is smaller than the former.) The average mean sizes of the confidence regions based on 12 model trees are shown on the x-axis in Figure 3.3 for both 80% and 95% levels and sequence lengths of 500 and 1000. The solid line parallel to x-axis shows the nominal coverage. The lower bound of the confidence interval of the nominal coverage is shown by the dotted line. Any coverage beyond the lower bound of the confidence interval may indicate the over-confidence of the method since the probability that it happens is smaller than 0.05. For each method, the 12 ‘+’s indicate the coverage of each model tree. Note that in some cases the coverages are the same for different model trees so that there are less than 12 distinct ‘+’s. When the coverages are below the lower bound, ‘+’ is replaced by the labels of the model trees. With the smallest PMLTs among 12 model trees, Trees H, J, K and L appear most frequently below the dotted lines.
Figure 3.3: The simulation results of 5-taxon trees.
Large Number of Taxa

When the number of taxa becomes large, the size of the confidence region depends on the number of candidate trees and on the method of choosing them. A single data set was simulated from each of the 3 model trees with large number of taxa in order to get a basic idea of the behavior of the methods. At first 100 trees with the highest likelihoods were selected by a heuristic search with neighbor joining starting trees followed by nearest-neighbor-interchanging (NNI) branch swapping as the candidates. However they might not be the 100 best trees globally and some of them were partially star trees (some internal branch lengths are 0). If a partially star tree is not rejected, all the binary trees nested in such structure should be included in the confidence region as well. Although the number of trees in the confidence region is not the true size in this section, a small number of trees still indicates a small size. In Table 3.3, the number of trees in the confidence region was counted as the amount of candidate trees with $P$-values larger than 0.05, except for the LW test. Each tree in the confidence region with zero internal branch lengths was counted as 1. Note that the true tree was covered by the confidence regions in all these three examples.

<table>
<thead>
<tr>
<th>No.Taxa</th>
<th>No.trees</th>
<th>SDPB</th>
<th>SOWH</th>
<th>SDNB</th>
<th>SSWOH</th>
<th>GLS</th>
<th>AU</th>
<th>SH</th>
<th>LW</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>100</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>15</td>
<td>25</td>
<td>13</td>
<td>34</td>
<td>10</td>
</tr>
<tr>
<td>15</td>
<td>100</td>
<td>5</td>
<td>5</td>
<td>8</td>
<td>11</td>
<td>37</td>
<td>13</td>
<td>33</td>
<td>8</td>
</tr>
<tr>
<td>20</td>
<td>100</td>
<td>23</td>
<td>23</td>
<td>33</td>
<td>46</td>
<td>95</td>
<td>38</td>
<td>60</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3.3: The number of trees in 95% confidence regions of SDPB, SOWH, SDNB, SSWOH, GLS, AU, SH and LW for 10-, 15- and 20-taxon trees.

Basically the behavior of the methods follows the same pattern as for 5-taxon trees. Still the SOWH, SDPB and SDNB tests give us the smallest size of the confidence regions. But there were some exceptions. The number of trees in the confidence region of the SSWOH test became larger than the AU test in these three examples. The GLS test behaved strangely, the number of trees included was even larger than the SH test when the number of taxa increases.

For the SH and AU tests which involve tree comparisons in the null hypothesis, we also searched 500 trees with highest likelihoods by NNI branch swapping for the
20-taxon example. The $P$-values of the SH test increase with the increasing number of the tested trees as many authors already noticed (Strimmer and Rambaut 2001).

The AU test is much less sensitive to this variation. The $P$-values of the true tree for the AU test in both situations are the same, which is 0.43. But the $P$-value of the AU test changes when another 100 best trees were built as the candidates by nonparametric bootstrap instead of NNI branch swapping for the 20-taxon data set. The ML tree of the replicate data sets were recorded as the tested tree during the bootstrap until we obtained 100 distinctive ones. In this case, the largest difference of log-likelihoods between 100 trees is 12.8 but was 72.4 by NNI branch swapping, which means that the trees generated by nonparametric bootstrap are closer to the ML tree. (The ML tree picked by the two methods are same.) The corresponding $P$-value changed to 0.479. So for a given tree, the AU test does give different results depending on the set of candidate trees.

The likelihood weight in the LW test is not the likelihood weight when we only consider some of the possible trees. The performance of the LW test is stable when the number of the tested trees from NNI branch swapping changed from 100 to 500. And the number of trees in the confidence region became relatively smaller in the 20-taxon example even compared to the SDNB test. But there is a potential problem. The likelihood weight depends on not only the likelihood itself, but also the number of tested trees. When the number of tested trees is large, the trees with lowest likelihoods are always rejected, no matter how close their likelihoods are to the ML tree. It may cause low coverage if the true tree is one of them.

Because of the issues mentioned for the SH, AU and LW tests, we only compared the coverage of the other tests for the large trees. Again 100 trees with highest likelihoods by NNI branch swapping were selected as the candidates for each original data set. Table 3.4 lists the simulation results. The coverages of the SOWH, SDPB and SDNB tests were acceptable, which were around 0.95. The coverage of the SSOWH test was always 1. Conversely, the coverage of the 20-taxon tree went down to 0.84 for 95% confidence region for the GLS test. We also used a 30-taxon tree as the model tree to check the coverage of the GLS test and obtained coverage of 66% for 95% confidence region. The fact that these two observed coverages went beyond
the lower bound of the 95% confidence interval of the nominal coverage suggests that
the GLS test for large number of taxa is not reliable.

<table>
<thead>
<tr>
<th>No. Taxa</th>
<th>SOWH 95%</th>
<th>SOWH 80%</th>
<th>SDPB 95%</th>
<th>SDPB 80%</th>
<th>SDNB 95%</th>
<th>SDNB 80%</th>
<th>SSOWH 95%</th>
<th>SSOWH 80%</th>
<th>GLS 95%</th>
<th>GLS 80%</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.93</td>
<td>0.85</td>
<td>0.93</td>
<td>0.85</td>
<td>0.93</td>
<td>0.87</td>
<td>1</td>
<td>1</td>
<td>0.93</td>
<td>0.78</td>
</tr>
<tr>
<td>15</td>
<td>0.94</td>
<td>0.88</td>
<td>0.94</td>
<td>0.88</td>
<td>0.94</td>
<td>0.82</td>
<td>1</td>
<td>1</td>
<td>0.94</td>
<td>0.86</td>
</tr>
<tr>
<td>20</td>
<td>0.92</td>
<td>0.85</td>
<td>0.92</td>
<td>0.84</td>
<td>0.96</td>
<td>0.81</td>
<td>1</td>
<td>1</td>
<td>0.84</td>
<td>0.74</td>
</tr>
</tbody>
</table>

Table 3.4: The coverage of the confidence region with sequence length of 3000 for 10-, 15- and 20-taxon trees.

3.3 Discussions

3.3.1 Comparison of the Tests

The discussions are mostly based on the simulation results of 95% confidence region of 5-taxon trees (see Figure 3), combined with the results from the large number of taxa. The results of 80% confidence region serve as reinforcement of our opinions.

The SDPB, SDNB and SOWH tests behave similarly. Many results have shown that the confidence region only contained the ML tree when the SOWH test was applied to real data even with high confidence level (e.g. Goldman, Anderson and Rodrigo 2000; Buckley 2002). But ignoring the problem of model misspecification, the performance of the SOWH test is the best. Although the coverages were below the nominal level sometimes, the smallest coverage was 91% for confidence level 95%, which was not outside the sampling interval. However it is very time consuming to test many trees one by one. The SDPB and SDNB tests solve this problem by using one bootstrap distribution for all trees to be tested.

The SDPB test creates the smallest mean size confidence region but there are some issues about the coverage. For instance with sequence length of 500, the coverage can be as low as 87% for 95% confidence region. The SDNB test has higher coverage but with a larger size of the confidence region. Sometimes its coverage was lower than the nominal coverage but seldom below the lower bound. Compared to the SOWH and SDPB tests, the SDNB test is not as dependent on the model assumptions.
The coverages of the SH and AU tests exceed the nominal coverage for all situations of the 5-taxon trees in Figure 3. The sizes of the confidence regions based on these two methods were the largest among the eight tests with the size of the SH region being noticeably larger. Since they were derived as ways of adjusting for the multiple comparisons with more than two trees, it may seem possible that their performance would be more comparable to the other tests when m=2. However this did not turn out to be the case in the simulations we considered. For example, when 500 sites were generated from tree G and the tests were applied with an alternative tree that switched taxon labels 1 and 4 (see Figure 1), the average sizes of the confidence regions were 2 and 1.89 for the SH and AU tests, and 1.58 for the SDNB test. Note that since the largest size of the confidence region is 2 for two candidate trees, the difference between these sizes seems not be significant but still indicates the conservative performance of the SH and AU tests. Comparison with confidence regions when m=15 suggests that the conservative performance of the SH and AU tests is not due simply to multiple testing issue.

The LW test tended to give smaller confidence regions than the AU test when the number of taxa is small. But for large numbers of taxa, its coverage and size are related to the trees in the testing pool as we have explained in section 3.2.4, which is not a good feature for implementation.

The SSOWH test always has coverage above the nominal level but with much smaller size of the confidence region than the SH, AU or LW test for 5-taxon tree. Although on average there were two trees less for the SSOWH test under the same condition than the LW test, we can not recommend it as a good choice since the size of the confidence region became larger than the AU test for the 20-taxon tree. For a 4-taxon tree, the star tree is the only common structure among the 3 possible unrooted trees and can be considered as the boundary of the tree space. But for 5-taxon tree, ((1,2)(3,4,5)) is the true boundary of ((1,2)(3,(4,5))), ((1,2)((3,4),5)) and ((1,2)((3,5),4)). By shortening the internal branch between (1,2) and (3,4,5), we can finally achieve the star topology. The star tree as the boundary makes $H_0$ hard to reject. It gets worse with increasing number of taxa.

As the only test based on distance methods, the GLS test worked reasonably well
for 5-taxon trees. Sizes of confidence regions tended to be smaller than for the AU and SH tests but larger than for the bootstrap-based tests. Problems occurred with large numbers of taxa. The coverage was much less than the nominal level and the size of the confidence region could be extremely large. There are two possible reasons for this behavior. Although the pairwise distances have an asymptotic multivariate normal distribution, it is unclear how large sequence lengths have to be for the asymptotic approximation to be good with large numbers of taxa. The second reason is that with larger numbers of taxa, the estimated covariance matrix is more likely to be close to singular. Since the GLS test statistics are sometimes undefined in such cases, a modified routine was used whenever some of the eigenvalues in the estimated covariance matrix were less than $10^{-6}$. This modification was based on the idea that generalized least squares can be interpreted as weighted least squares for a set of approximately independent linear transformations. The modification ignores those linear transformations, with estimated variance near 0, that create problems when the covariance matrix is near singular. Although this avoids difficulties with undefined GLS test statistics, some information is lost in ignoring these linearly transformed distances.

Some authors concluded that the difficulty of phylogenetic analysis increases with the increasing diameter of the underlying tree, especially for trees with small number of taxa (Huelsenbeck and Hillis 1993). The diameters of the model trees in the paper of Zwickl and Hillis (2002) ranged from 0.1 to 0.45. In our simulations of five taxa, even the smallest diameter was 1.8, which meant we considered very hard conditions for 5-taxon trees. The acceptable coverages of the SDPB and SDNB tests for large number of taxa are partly due to the small diameter of the model trees, which are less than 0.45. The PMLT is a scale to reflect the difficulty of phylogenetic analysis in the simulation. Even for the very large diameters of the 5-taxon trees, the coverages of the SDPB and SDNB tests are reasonable if the PMLT is larger than 0.6.

When the true tree has star topology, all the candidate trees are acceptable for the observations, which is the underlying assumption of the SH and SSOWH tests. All the other tests undercover at varying levels. For instance, the coverage of the SDNB, SOWH and SDPB tests went down to 87%, 85% and 80% for 95% confidence region
by shrinking the two internal branches of tree G to 0.001. This example indicates a possible reason for the small confidence region of the SOWH test of observed in reality. While the SOWH test generally performed well, this is a region of tree space where it could be expected to have some difficulties. Because branch lengths are bounded below by 0, the estimated branch lengths for any hypothesized tree are much more likely to be larger than the true branch lengths. The SOWH test is thus simulated from a tree with more “structure” than the true tree. The level of uncertainty reflected in the estimated distribution of the likelihood ratio statistic is less than it would have been if the actual branch lengths were used in simulations. The consequence, as the simulations confirm, is undercoverage when true branch lengths are close to 0. While these are cases where the SOWH test will not perform well, phylogenies near the star phylogeny may not be common in real problems of biological interest but could arise in some special situations.

Note that the performances of the seven likelihood-based methods depend on the number of the trees tested for large number of taxa since the maximum likelihood increases with increasing number of tested trees. The GLS test is not affected by this parameter.

3.3.2 Effects of Factors

Sequence length, tree shape and branch lengths are the factors to influence the difficulty of the phylogenetic analysis. We present their effects based on the simulation results of small number of taxa.

As expected, increasing the sequence length enables us to estimate the tree more precisely and obtain smaller confidence regions. The average mean size of the confidence regions with sequence length of 1000 are almost three trees less than with sequence length of 500 in Figure 3.3.

Tree shape is determined by the relative positions of long and short branches. The diameters of the three trees are different even when they have the same long and short branch lengths. Since there are two internal branches between the two long branches in the left tree, its diameter is the largest, followed by the middle and right one. We found that the sizes of the confidence regions of the data sets simulated from the left
two trees in Figure 3.1 were smaller than the right one. For example, the simulation results of the SDPB test from tree C, G and K with sequence length of 500 is given in Table 3.5. Tree C corresponds to the left model tree in Figure 3.1 with PMLT as 0.63, G to the middle with PMLT as 0.67 and K to the right with PMLT as 0.42. All the 15 tree topologies corresponding to the index in the table is presented in Figure 3.4.

![Figure 3.4: The 15 possible tree topologies of 5 taxa](image)

Generally speaking, the trees with same subtrees as the model tree are more frequently included in the confidence region, such as tree 1, 2, 9 and 15 (tree 5 has the same topology as the model tree). Tree C arose in 94 of the confidence regions
obtained through simulation. Trees that differed from C by a single nearest neighbor interchange (tree 1, 2, 9 and 15) appeared 40-46 times. The trees with long branches together occurred 25, 28 and 18 (tree 6, 7 and 10 with (Taxon2, Taxon4) as the common subtree) times which was consistent with the numbers of times (18-28) that other trees differing from C by more than one nearest neighbor interchange appeared. More interestingly, when the two taxa (taxon 2 and 3) with long branches are together in tree K, trees with the same subtree (tree 1 and 2) occurred even less than in tree C and G while tree 9 and 15 occur much more to lead to a larger size. We are surprised that long branch attraction (LBA) did not occur in this situation. However, there is no model-misspecification in the simulation. Many of the recorded cases of LBA involve model misspecification. And LBA has been reported as a small sample bias, but internal edge lengths are usually small when it occurs. It may be that the situation was not extreme enough for it to be a problem.

<table>
<thead>
<tr>
<th>Tree</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>size</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>40</td>
<td>45</td>
<td>22</td>
<td>20</td>
<td>94</td>
<td>25</td>
<td>28</td>
<td>28</td>
<td>44</td>
<td>18</td>
<td>24</td>
<td>20</td>
<td>21</td>
<td>18</td>
<td>46</td>
<td>4.93</td>
</tr>
<tr>
<td>G</td>
<td>49</td>
<td>45</td>
<td>10</td>
<td>10</td>
<td>93</td>
<td>11</td>
<td>17</td>
<td>14</td>
<td>51</td>
<td>11</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>11</td>
<td>54</td>
<td>4.06</td>
</tr>
<tr>
<td>K</td>
<td>25</td>
<td>26</td>
<td>21</td>
<td>22</td>
<td>93</td>
<td>21</td>
<td>35</td>
<td>31</td>
<td>83</td>
<td>21</td>
<td>21</td>
<td>23</td>
<td>35</td>
<td>36</td>
<td>82</td>
<td>5.75</td>
</tr>
</tbody>
</table>

Table 3.5: The number of appearance of each candidate tree in 95% confidence regions for 100 data sets simulated from tree C, G and K respectively.

The branch length gives the expected substitutions in a certain time. In order to sort out the effect of the relative branch lengths, it is instructive to look at the results for trees I & K which have the same diameter (1.8) and the same long branch lengths, but the short branch lengths are 0.03 for I and 0.06 for K. The mean sizes of 95% confidence regions from the SOWH test with 500 sequence length were 6.6 and 3.71 respectively. (The coverages of the SOWH test for these two trees were both larger than 0.95.) Obviously the relatively longer short branch in Tree K was the reason for the smaller size. We enlarged the short branches of the tree structure on the right in Figure 3.1 and listed the values of the PMLT out of 100 replicates in Table 3.6. When the tree structure became fully symmetric with 0.9 as the short branch lengths, the PMLT went up to 99 even when the diameter increased to 3.6. When we broke the symmetry by increasing the length of branch a, d, e, f and g beyond 0.9, the PMLT
decreased. Note that the changing of the short branch lengths also changes the shape of the tree.

<table>
<thead>
<tr>
<th>tree structure</th>
<th>b, c</th>
<th>a, d, e, f, g</th>
<th>PMLT</th>
<th>diameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>partly symmetric</td>
<td>0.9</td>
<td>0.03</td>
<td>42</td>
<td>1.8</td>
</tr>
<tr>
<td>(the right tree</td>
<td>0.06</td>
<td>0.56</td>
<td>56</td>
<td>1.8</td>
</tr>
<tr>
<td>in Figure 3.1.)</td>
<td>0.9</td>
<td>0.69</td>
<td>99</td>
<td>3.6</td>
</tr>
<tr>
<td></td>
<td>1.5</td>
<td>0.94</td>
<td>94</td>
<td>5.4</td>
</tr>
</tbody>
</table>

Table 3.6: The values of the PMLT out of 100 replicates when the short branch lengths increase for the tree structure on the right in Figure 3.1.

3.3.3 Parametric versus Nonparametric Bootstrap

There are two kinds of bootstrap: parametric and nonparametric. In phylogeny, parametric bootstrap requires a substitution model along with the estimated parameters. If the model is misspecified, this may introduce bias into the bootstrap results (Buckley 2002). Nonparametric bootstrap is more popular since it is independent of the model in terms of bootstrap sampling.

The 95th percentiles of the SDNB and SDPB bootstrap distributions of the likelihood ratio statistic estimate the 95th percent of the actual distribution. For a given true tree, the 95th percentile of the actual distribution can be approximated as the 95th percentile of likelihood ratios obtained by repeated simulating data sets from the true tree. We did this for three trees at different levels of PMLT: 0.94 (tree A), 0.63 (tree C) and 0.25 (tree L). In each case, 5000 data sets with sequence length of 500 were generated. This gave actual 95th percentiles 0.257, 1.39 and 2.565 for the three trees.

To investigate the biases and variances due to the differing forms of bootstrapping in SDNB and SDPB, we selected 100 data sets from the 5000 and applied the procedures with RELL resampling. For each tree and both of the SDNB and SDPB tests this gave 100 estimated 95th percentiles. The result is given in Table 3.7.

The standard deviations reflect two sources of variations: the bootstrap methods and the number of bootstraps. Since we are interested in the former one, the same number of bootstraps were used in the three techniques in order to make them
<table>
<thead>
<tr>
<th></th>
<th>SDNB</th>
<th>RELL</th>
<th>SDPB</th>
<th></th>
<th>SDNB</th>
<th>RELL</th>
<th>SDPB</th>
<th></th>
<th>SDNB</th>
<th>RELL</th>
<th>SDPB</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean</td>
<td>1.7</td>
<td>2.23</td>
<td>0.71</td>
<td>tree A</td>
<td>0.257</td>
<td></td>
<td></td>
<td></td>
<td>tree C</td>
<td>1.39</td>
<td></td>
</tr>
<tr>
<td>SD</td>
<td>2.25</td>
<td>2.41</td>
<td>0.79</td>
<td></td>
<td>2.57</td>
<td>2.09</td>
<td>0.78</td>
<td></td>
<td>1.42</td>
<td>1</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Table 3.7: The means and the standard deviations (SD) of the cutoff values in the SDNB, SDNB with RELL and SDPB tests.

comparable, which is 100. The table shows that the parametric bootstrap provides smaller mean cutoffs than nonparametric bootstrap. With the decreasing PMLT, the true cutoff increases. Though the cutoffs from the SDPB test increase as well, their mean underestimates the true cutoff of tree L. Conversely the SDNB test provides better estimate in this situation even though it overestimates the true cutoffs of tree A and C. Note that the nonparametric bootstrap makes the model tree less possible as the ML tree in the replicates than the original data set. We notice that the standard deviation of the nonparametric bootstrap is always larger than the parametric one. It means that the parametric bootstrap provides more stable cutoffs than the nonparametric one.

Obviously, the performances of the bootstrap methods are related to the PMLT as well. The nonparametric bootstrap tends to be conservative. Note that the underestimate or overestimate of the true cutoff doesn't necessarily mean high or low coverage. It is possible that small cutoff corresponds to smaller test statistic value. The results in this part only show the behavior patterns of the bootstraps.

3.4 Conclusions

In evaluating the various test methods, we considered the coverage and the size of the confidence regions based on simulations. Generally high coverage causes large size and small size leads to low coverage. The good method gives a small confidence region with reasonable coverage. Overall the SOWH test performed best during the simulation, with reasonable coverage and relative small size. But in this chapter, we only examine the the efficiency of various tests without considering the model assumption violation. The sensitivity of the SOWH test to model misspecification is the major reason for its lower coverage (Buckley 2002). So we don't recommend
it unless the substitution model is developed to be closer to the real world. The SDPB test is not recommended for the same reason, which is expected to have lower coverage than the SOWH test in the same situation. Though the SDNB test loses some power to detect the wrong tree in the confidence because of the nonparametric bootstrap, it is less dependent on the model assumption. And it avoids the time-consuming problem in the SOWH test when the number of candidate trees is huge. We recommend it when the power of the test is highly required.

Since the SSOWH, LW and GLS tests perform unstably, they are not recommended for large number of taxa. We dont recommend the SH and AU tests for small number of taxa since the large size reflects the lack of power to detect the wrong trees. As a popular method, the AU test is conservative and sensitive to the set of candidate trees. But it is not seriously misleading for large number of taxa. The AU test is safe to use in order to control the coverage strictly.

Note that only the binary trees were considered as the model trees in our simulation. With the star trees as the models, all the candidate trees equally support the data. Except for the SH and SSOWH tests, all the other tests tend to undercover in this situation especially the SOWH, SDNB and SDPB tests.

The performance under model misspecification is not discussed here either. Under such situations, the performance of these methods depends on how far way the substitution model is from reality. Model misspecification is an important issue but it involves a whole span of other issues taking us beyond the scope of this chapter.
Chapter 4

Testing Splits in Phylogenetic Trees

In this chapter, the confidence of a prespecified split in a tree is of interest instead of the whole tree structure. It is interesting to address this issue as a hypothesis test problem and use standard test methods on it.

We develop a simple and valid method to test the partially fixed tree structures. Our approach will be to decompose the original hypothesis $H_0$ into a number of individual hypotheses $H_{0i}, i = 1, \ldots, M$, in which the tested trees have fixed structures due to the null hypothesis, where $M$ is the total number of individual hypotheses needed. Some existing test methods can then be adopted for these hypotheses to find $P$-values. A conclusion about the original hypothesis can be made through dealing with these $P$-values. A true null hypothesis is a true statement about a subset of taxa which constitute a split of the true tree and this split can be isolated from the other part of the structure by breaking only one internal branch. For example, the constraint that taxa 4 and 5 are in the same split is true in the 10-taxon tree in Figure 4.1. However since we must break two internal branches to make the split of (4, 5, 6) isolated from the whole structure, the corresponding constraint is false.

The rest of the chapter is organized as follows. At first the relationship between the original hypothesis and a number of individual hypotheses is introduced. In each individual hypothesis, a 4-taxon tree is tested, which has fixed structure since two taxa are neighbors under the null hypothesis. Then four test methods, namely AU, GLS, SSOWH and SDNB, are reviewed based on 4-taxon trees for these individual hypotheses. Three methods are used to cope with the multiple $P$-values. Two of them are the widely used false discovery rate (FDR) controlling procedure (Benjamini and Hochberg 1995) and a Chi-square approximation to a function of independent $P$-values, which are the main focus. A traditional familywise error rate (FWER)
controlling procedure, the Hochberg's method is included only for comparison. Subsequently, all combinations of the tree test methods and the multiple test procedures are compared for some constraint tests on a simulated 10-taxon tree. These combinations are assessed based on type I error rate and power. As the best choice of the test method, the AU test is then applied on the 46 eubacterial and eukaryotic homolog data combined with multiple test procedures.

4.1 The rationale of the test procedure

Suppose in a $S$-taxon tree, a split with $S_0$ taxa is being tested, then the left $S - S_0$ taxa will be formed into another group naturally. The overall null hypothesis can be written as:

$$H_0: \text{The } S_0 \text{ (or the remaining } S - S_0 \text{) taxa are in the same split}$$  \hspace{1cm} (4.1)

First consider the simplest meaningful case where $S_0 = 2$, i.e. the two taxa are neighbors under $H_0$. In other words, the two taxa are closer to each other than to any others. Under the null hypothesis, these two taxa will be neighbors for any 4-taxon subset which includes them as members. If $H_0$ is false, there is at least one
taxon existing in between them in the true tree and thus in some of the 4-taxon trees. Inversely, if two taxa are neighbors in all the possible \((S-2)\) 4-taxon trees, they must be neighbors in the \(S\)-taxon tree since none of the other taxa can break the alliance between them. A simple example is given to demonstrate this idea. In Figure 4.2, taxa 1 and 2 are neighbors in the 5-taxon tree. It implies that any 4-taxon tree that includes them should have them as neighbors, as shown in the second row. But the topologies in the second row can only tell that taxon 1 and 2 are neighbors in a 5-taxon tree. The topology among taxa 3, 4 and 5 is not of concern in the hypothesis.

Figure 4.2: An example for demonstrating the basic idea of the constraint test.

The situation is more complicated when \(S_0 > 2\) due to the unknown tree structure
within this split. The idea is still based on that the taxa within this split are closer to each other than to taxa outside this split. Thus under $H_0$, any 4-taxon tree which includes two taxa within the split and the other two from outside the split has a fixed structure with the former two taxa as neighbors. For simplicity of description, we call the two taxa chosen from the tested split in the null hypothesis as taxon 1 and 2 in a 4-taxon tree. They are neighbors under $H_0$, as shown in Figure 4.3.

![Figure 4.3: The structure of the tested tree under $H_{0i}$ with taxon 1 and 2 from inside of the split and the other two from outside of the split.](image)

We denote this tree structure as $\tau_1$ and the other two possible structures as $\tau_2$ and $\tau_3$. There are $M = \binom{S_0}{2} \times \binom{S-S_0}{2}$ possible 4-taxon combinations to be tested as individual null hypotheses. Testing the original hypothesis (4.1) is equivalent to testing:

$$H_{0i} : \text{The } ith \text{ 4-taxon tree has } \tau_1 \text{ as its true topology, for all } i = 1, \ldots, M$$

(4.2)

The original hypothesis (1) is accepted if all $H_{0i}$ ($i = 1, \cdots, M$) are accepted and rejected if at least one $H_{0i}$ ($i = 1, \cdots, M$) is rejected. Thus testing the original hypothesis becomes a familywise error rate (FWER) controlling multiple tests. Note that when at least one taxon outside the split breaks the alliance of the taxa within the split, there are often more than one individual hypothesis among $H_{0i}$ should be rejected. Thus the high power of such test procedure is expected.
4.2 Tree Test Methods for Individual Hypotheses

Since the individual tests are the ordinary tests with a fixed tree structure in the hypothesis, the existing test methods can be applied to provide \( P \)-values. Among these methods, the SH test is too conservative since the least favorable configuration assumed in the hypothesis corresponds to a specific hypothesis very difficult to reject; the SOWH and SDPB tests are acceptable only when there is no model mis specification (Shi et al. 2005). Here four other methods, namely AU, GLS, SSOWH and SDNB, are considered for the individual hypotheses. A simple review of these methods for a 4-taxon tree is given below.

As the the widely used test method nowadays, the hypothesis of the AU test is stated in terms of the expected log-likelihood of topology \( \tau_i \), denoted by \( \mu_i, i = 1, 2, 3 \). The hypothesis for testing \( \tau_i \) is

\[
H_0 : \mu_i \geq \mu_j, \quad j = 1, 2, 3
\]

The AU test controls the type I error strongly with relatively weak power. As an important feature, the accuracy of the \( P \)-value in the AU test can be increased by increasing the number of bootstraps.

In the GLS test, the ML pairwise distances \( Y \) contains six elements that can be expressed as the sums of the branch lengths on the path between the pairs of taxa. The design matrix \( X \) expressing the tree topology in Figure 4.3 is fixed in all the individual tests. The following equation shows the relationship between \( Y \), \( X \) and \( T \) under \( H_0 \), where \( T \) is a vector of branch lengths estimated to minimize \( (Y - XT)^T V^{-1} (Y - XT) \), where \( V \) is the covariance matrix of the pairwise ML distances.

\[
\begin{pmatrix}
\text{Taxon1-2} \\
\text{Taxon1-3} \\
\text{Taxon1-4} \\
\text{Taxon2-3} \\
\text{Taxon2-4} \\
\text{Taxon3-4}
\end{pmatrix}
\begin{pmatrix}
1 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 \\
1 & 0 & 0 & 1 & 1 \\
0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 & 0
\end{pmatrix}
\begin{pmatrix}
t_1 \\
t_2 \\
t_3 \\
t_4 \\
t_5
\end{pmatrix}
\]
Since the ML pairwise distances are asymptotically multivariate normal distributed, under $H_0$, $(Y - XT)^T V^{-1} (Y - XT)$ asymptotically follows a Chi-square distribution with 1 degree of freedom. By using this asymptotic result to avoid bootstrap, the GLS test has some computational advantage over the other methods, which makes it more desirable in settings with a large number of individual hypotheses.

In the SSOWH test, the test statistic is the difference of log-likelihood between the ML tree and $\tau_1$, whose distribution is constructed by a parametric bootstrap. The replicate data sets are simulated under a star topology with ML branch lengths estimated from the original observations. The replicate of the test statistic is the log-likelihood difference between the ML tree and the tested tree of the replicate data set. It is shown that with large number of taxa, the SSOWH test does not perform well when the underlying true tree topology is far away from the star topology. Since only 4-taxon trees are considered here, it is not a problem to treat the star topology as their boundary.

The same test statistic is used in the SDNB test. But its distribution is constructed by nonparametric bootstrap. According to the bootstrap theory, the bootstrap replicate of the test statistic is the log-likelihood difference between the ML tree of the replicate data set and the ML tree of the original data set with other free parameters estimated in the replicate data set. The SDNB test may raise a problem about type I error if there are short internal branches in the true tree.

The accuracy of the $P$-values from the last two methods closely relates to the number of bootstraps. If the number of bootstrap is 100, the $P$-value can only be accurate up to two decimals. The important issue about these two tests is the heavy computational burden to obtain high accuracy of the $P$-values for each hypothesis and the large number of individual hypotheses.

4.3 Procedure

We summarize the procedure of testing all the possible 4-taxon trees under the hypothesis as below:

1. Suppose there are $S_0$ taxa within the split under constraint. Select two taxa from $S_0$ taxa inside the split as taxon 1 and 2 and the other two outside the split and
combine their sequences to form a subdata set.

2. Implement the chosen test method for corresponding \( H_0i \) and record the \( P \)-value.

3. Go back to step 1 until all the 4-taxon combinations are considered. It means that we must test all of \( M \) combinations, where \( M = \binom{S_0}{2} \ast \binom{S - S_0}{2} \).

There are three advantages in this proposed test procedure. At first, the tree topology need not be estimated for any of these 4-taxon trees. As a tested topology, it has major advantage over the ML tree under the constraint. Secondly, by using 4-taxon trees in individual hypotheses, the number of individual hypotheses is minimized. i.e. \( \binom{S_0}{2} \ast \binom{S - S_0}{2} \leq \binom{i}{2} \ast \binom{j - S_0}{2} \), for any \( i, j \geq 2 \). It reduces the problems in multiple comparisons and avoids intensive computations to the best extent. At last, based on the simulation results in Shi et al. (2005), for the asymptotic results to be valid, trees with large number of taxa need long sequences in the AU and GLS tests and thus involve intensive computation. However they work well for 4-taxon trees with relative short sequence lengths.

### 4.4 Multiple \( P \)-values

After obtaining \( P \)-values of individual hypotheses, a method for making statements on overall hypothesis is needed. Let \( H_0 = \{ H_{01}, ..., H_{0M} \} \) be a set of null hypotheses with corresponding test statistics \( T_1, ..., T_M \), and \( P \)-values \( P_1, ..., P_M \), where \( M = \binom{S_0}{2} \ast \binom{S - S_0}{2} \). \( H_0 \) is accepted when none of \( H_{0i} \), \( i = 1, ..., M \) is rejected. Three methods are examined here to make inferrence on the original hypothesis. The first two procedures (the false discovery rate (FDR) controlling procedure and Hochberg's procedure) are from general multiple comparison point of view and the third one makes statements by a Chi-square approximation procedure based on independent \( P \)-values.

#### 4.4.1 The FDR Controlling Procedure

Benjamini and Hochberg (1995) proposed the idea to control the false discovery rate (FDR) instead of familywise error rate (FWER) in multiple tests. Consider the problem of testing simultaneously \( M \) null hypotheses with \( M_0 \) of them being true. \( R \)
is the number of hypotheses rejected and $R_0$ is the number of false rejection among them. The situation is summarized in Table 4.1.

<table>
<thead>
<tr>
<th></th>
<th>non-significant</th>
<th>significant</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>True null hypotheses</td>
<td>$U_0$</td>
<td>$R_0$</td>
<td>$M_0$</td>
</tr>
<tr>
<td>False null hypotheses</td>
<td>$U_1$</td>
<td>$R_1$</td>
<td>$M - M_0$</td>
</tr>
<tr>
<td></td>
<td>$M - R$</td>
<td>$R$</td>
<td>$M$</td>
</tr>
</tbody>
</table>

Table 4.1: Number of errors committed when testing $M$ null hypotheses

The FDR can be viewed as expectation of the proportion of errors committed by false rejection of the null hypotheses:

$$\text{FDR} = E(Q) = E(R_0/R)$$

Define $Q = 0$ when $R = 0$, as no error of false rejection can be committed. In FDR controlling procedure, $P_{(1)}, ..., P_{(m)}$ are the increasingly ordered $P$-values for $H_0 = \{H_{(01)}, ..., H_{(0m)}\}$. All $H_{(0i)}$ for $i = 1, ..., I$ are rejected, where $I$ is the largest $i$ for which $P_{(i)} \leq \frac{i}{m} q^*$, where $q^*$ is the desired level of FDR.

This procedure controls the FDR for both independent and positively dependent test statistics (Benjamini and Liu 1997; Benjamini and Yekutieli 1999). If all of the null hypotheses are true, there are only two possible values for $Q$: $Q = 0$ when $R = 0$ and $Q = 1$ when $R \geq 1$. Then the FDR is equivalent to the FWER:

$$\text{FDR} = E(Q) = 1 \times P(R \geq 1) + 0 \times P(R = 0) = P(R \geq 1) = \text{FWER}$$

Therefore controlling the FDR implies controlling the FWER. When the constraint is not true, FDR controlling procedure is more powerful than the general FWER controlling procedure. The overall hypothesis is rejected if there is at least one $H_{(0i)}$ rejected or equivalently if $I \geq 1$.

### 4.4.2 The Chi-square Approximation Procedure

Generally if the distribution of the test statistic is continuous, under the true null hypothesis, the $P$-value is uniformly distributed between 0 and 1. Let $Y = -\log P$, where $P$ denotes the $P$-value with a uniform distribution. Then the probability density function of $Y$ is

$$p_Y(y) = e^{-y}, \quad y \geq 0$$
which is consistent with the density function of the gamma distribution \( \Gamma(1, 1) \). From the following two theorems:

1. \( X \) is distributed \( \Gamma(\alpha, \lambda) \) if, and only if, \( \lambda X \) is distributed \( \Gamma(\alpha, 1) \).

2. If \( X_1 \) and \( X_2 \) are independent random variables with \( \Gamma(\alpha_1, \lambda) \) and \( \Gamma(\alpha_2, \lambda) \) distributions, respectively, then \( Y = X_1 + X_2 \) has \( \Gamma(\alpha_1 + \alpha_2, \lambda) \) distribution.

It can be easily shown that the statistic \( -2 \sum_{i=1}^{M} \log R_i \) has a Chi-square distribution with \( 2M \) degrees of freedom if the \( P \)-values are independent.

It is likely that the \( P \)-values resulted from these 4-taxon trees are positively correlated since all the taxa are used more than once to construct these 4-taxon trees. But from the simulation examples we found their correlations are typically very small. This method is thus included here. Note when bootstrap is used in calculating \( P \)-values in the tree test methods (AU, SOWH and SDPB), \( P \)-value can only be accurate up to some fixed decimals depending on the number of bootstraps used. It is possible that the resultant \( P \)-value is zero, which in fact means the \( P \)-value is smaller than the smallest possible \( P \)-value we would get from bootstraps. In such cases, theoretically it can’t be implemented because of the infinite logarithm issues in this method. However in order to compare the result with FDR controlling procedure, we replace the zero \( P \)-value by the the midpoint between 0 and the smallest possible \( P \)-value from the tree tests. We reject the overall hypothesis if \( -2 \sum_{i=1}^{M} \log R_i \) is greater than the \((1 - \alpha)\) critical value of \( \chi^2(2M) \).

### 4.4.3 Hochberg’s Procedure

As the early popular method for conducting multiple tests, the Bonferroni inequality is often used to set an upper bound on the overall significance level \( \alpha \). The classical Bonferroni multiple test procedure is usually performed by rejecting \( H_0 \) if any \( P \)-value is less than \( \alpha/M \). Furthermore the specific hypothesis \( H_{0i} \) is rejected for each \( P_i \leq \alpha/M \). The Bonferroni inequality,

\[
Pr\{ \bigcup_{i=1}^{M} (P_i \leq \alpha/M) \} \leq \alpha
\]

ensures that the probability of rejecting at least one hypothesis when all are true is no greater than \( \alpha \). However the Bonferroni procedure controls FWER in such strong
sense that it is conservative and lacks power.

Hochberg proposed a new procedure in 1988 to control FWER with stronger power. The procedure is to get the largest $i$, denoted as $I$, for which $P_{(i)} \leq \frac{\alpha}{M+1-i}$. Then all $H_{(0)}, i = 1, \ldots, I$, are rejected. As a step-down procedure, it starts by examining the largest $P$-value $P_{(M)}$. If $P_{(M)} \leq \alpha$, then all hypotheses are rejected. If not, then $H_{(0M)}$ can not be rejected and one goes on to compare $P_{(M-1)}$ with $\frac{1}{2} \alpha$. If smaller, then all $H_{(0)}, i = M - 1, \ldots, 1$ are rejected. Otherwise $H_{(0, M-1)}$ cannot be rejected and one proceeds to compare $P_{(M-2)}$ with $\frac{1}{3} \alpha$ and so on until $P_{(1)}$. If $P_{(1)} > \frac{1}{M} \alpha$, then none of $H_{(0)}, i = M, \ldots, 1$ is rejected. Neither is the overall hypothesis $H_0$ with significance level $\alpha$.

4.5 Simulation

4.5.1 Models

Similar to the process of constructing the model tree in the previous chapter, one data set was simulated from that published 66-taxon tree (Murphy et al. 2001) with sequence length of 3000 at first. Then 10 taxa were selected from the 66 taxa and the ML tree under the constraints of (1, 2), (4, 5) and (8, 9) was constructed by PAUP* as the model tree. This model tree has structure the same as the tree in Figure 4.1. The internal branch lengths in this tree are no longer than 0.04 and some of them are close to 0, such as the internal branch between the subtree (8, 9) and taxon 10. Note that it is different from the 10-taxon tree in chapter 3.

The F84 model was used for simulation and analysis with base frequencies as $(\pi_a, \pi_c, \pi_g, \pi_t) = (0.37, 0.24, 0.12, 0.27)$ and transition/transversion ratio ($\kappa$) as 2.93. These parameters are the same as in chapter 3 for large number of taxa. The sequence length is fixed as 500 for simulation.

4.5.2 Methods

As previously, the AU and GLS tests were performed using CONSEL (Shimodaira and Hasegawa 2001) with default settings and the C program glsdna_eig (Susko 2003). The distribution of the test statistics in the SSOWH and SDNB tests were constructed
based on 100 bootstrap replicates.

After obtaining the $P$-values, the three multiple hypothesis testing methods made statements on the overall hypothesis. For the true constraint, the proportion of rejection can be viewed as FWER (FDR is equivalent to FWER in this situation); otherwise it indicates the power. The level of FWER and FDR are 0.05 for all the tests.

4.5.3 Results

The number of rejections of the constraint tests on 100 simulated data set is listed in Table 4.2. Our goal is to find the combination of the tree test method and the multiple test procedure which has strong power and controlled type I error. In Table 4.2, the results are given in two parts: true constraints in the upper part and false ones in the lower part. The first column shows the constraints being tested. The SDNB, AU, SSOWH and GLS tests are examined on each constraint. The number of individual hypotheses is 28 for $S_0 = 2$, 63 for $S_0 = 3$, 90 for $S_0 = 4$ and 100 for $S_0 = 5$. For each constraint test, we applied the FDR controlling procedure, Hochberg’s procedure and the Chi-square approximation method, denoted respectively by FDR, H and $\chi^2$ in Table 4.2. Since the multiple comparison methods are based on $P$-values, the character of the test methods are crucial in making the conclusion on $H_0$.

As the most powerful test method, the SDNB did not control the type I error well in the case of constraint (8,9). It is sensitive to the star topology and tends to reject more than it should. The short internal branch (0.0036) between subtree (8,9) and taxon 10 creates an analogous star topology, leading to large type I error in the SDNB test. Table 4.3 shows the number of rejections out of 100 under the same constraint with different internal branch lengths between the subtree (8,9) and taxon 10 under the SDNB + FDR combination. Obviously, in the range of 0.0036 and 0.036, the longer the internal branch is, the smaller the type I error is.

The SSOWH test has the highest power among the remaining methods. Although the SSOWH test uses the unrealistic underlying assumption in the parametric bootstrap which is that all the possible tree topologies equally support the observations, it didn’t cause serious problems for 4-taxon trees. However with its low type I error
| constraint | SDNB  | |  | AU  | |  | SSOWH  | |  | GLS  | |  |
|------------|-------|---|---|----|---|---|-------|---|---|---|---|
|            | FDR  | H | $\chi^2$ | FDR  | H | $\chi^2$ | FDR  | H | $\chi^2$ | FDR  | H | $\chi^2$ |
| (1,2)      | 1    | 1 | 1         | 0    | 0 | 0         | 0    | 0 | 0         | 0    | 0 | 13        |
| (4,5)      | 0    | 0 | 0         | 0    | 0 | 0         | 0    | 0 | 0         | 1    | 1 | 19        |
| (8,9)      | 18   | 18| 18        | 1    | 1 | 1         | 3    | 3 | 3         | 5    | 5 | 29        |
| (8,9,10)   | 0    | 0 | 0         | 0    | 0 | 0         | 0    | 0 | 0         | 4    | 3 | 37        |
| (7,8,9,10) | 0    | 0 | 0         | 0    | 0 | 0         | 0    | 0 | 0         | 2    | 2 | 37        |
| (8,10)     | 66   | 66| 66        | 4    | 4 | 4         | 31   | 31| 31        | 3    | 3 | 53        |
| (9,10)     | 63   | 63| 63        | 5    | 5 | 5         | 28   | 28| 28        | 7    | 7 | 52        |
| (6,7)      | 62   | 62| 64        | 5    | 3 | 3         | 51   | 51| 55        | 15   | 12| 62        |
| (3,6)      | 95   | 95| 96        | 17   | 13| 13        | 91   | 91| 93        | 46   | 34| 93        |
| (7,10)     | 100  | 100| 100       | 100  | 100| 100       | 100  | 100| 100       | 99   | 97| 100       |
| (7,8)      | 100  | 100| 100       | 95   | 88| 88        | 99   | 100| 100       | 100  | 100| 100       |
| (7,8,9)    | 100  | 100| 100       | 100  | 100| 100       | 100  | 100| 100       | 96   | 95| 100       |
| (7,8,10)   | 100  | 100| 100       | 89   | 75| 88        | 100  | 100| 100       | 98   | 97| 100       |
| (1,2,3,6,7)| 97   | 97| 98        | 83   | 83| 88        | 90   | 90| 91        | 39   | 24| 94        |

Table 4.2: The rejection times under the constraint based on the four test methods and three multiple hypothesis testing procedures.

<table>
<thead>
<tr>
<th>constraint</th>
<th>internal rejection times</th>
</tr>
</thead>
<tbody>
<tr>
<td>(8,9)</td>
<td>0.003</td>
</tr>
<tr>
<td></td>
<td>18</td>
</tr>
<tr>
<td>(8,9)</td>
<td>0.007</td>
</tr>
<tr>
<td></td>
<td>7</td>
</tr>
<tr>
<td>(8,9)</td>
<td>0.009</td>
</tr>
<tr>
<td></td>
<td>4</td>
</tr>
<tr>
<td>(8,9)</td>
<td>0.036</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

Table 4.3: The simulation result of the SDNB test and the FDR controlling procedure with increasing internal branch between subtree (8,9) and taxon 10.

and high power, the price to pay for the SSOWH test is the intensive computations for constructing the distribution of the test statistic, especially so when the number of individual tests and the bootstraps are large.

While the two multiple comparison methods didn't perform differently with the SDNB and SSOWH tests, the Chi-square approximation method outperformed the FDR procedure with stronger power when combined with the GLS and AU test. However the combination of the GLS test and the Chi-square approach procedure, didn't control type I error which was always larger than 0.1 and went up to 0.37 under the true constraint. The possible reason of this failure is due to the stronger correlations between the $P$-values resulting from the GLS test. Since all the taxa appear more than once in composing these 4-taxon trees, the independence assumption is violated
in all tests. But the dependence of $P$-values from other tree test methods is much weaker than that resulting from GLS.

By combining with the FDR controlling procedure, the type I error based on the GLS test was controlled along with similar power to the AU test. However the GLS tests provided weaker power for the constraint of $(1,2,3,6,7)$, while the performance of the AU test is comparable with the SOWWH and SDNB tests. There are five taxa in this constraint, which means that the $P$-values were compared to values as small as $0.05/100 = 0.0005$ with increment of 0.005 in the FDR procedure. The Chi-square distribution is an approximation in the GLS test with its accuracy depending on the sequence lengths. It seems that sequence length of 500 could not give enough accuracy for 5-taxon case and this accuracy is hard to improve.

We also found that the power changes under different false constraints with two taxa. When we need to remove more than two taxa to let the two taxa from the split be neighbours, it is easy to reject $H_0$. For example, it can be rejected almost 100 out of 100 times for the null hypothesis that taxon 7 and 8 are neighbors, since taxon 9 and 10 need to be taken out to make taxon 7 and 8 neighbors in the model tree. This is partially the reason for the relatively low power for the constraint tests of $(8,10)$ and $(9,10)$.

In summary, if the number of individual hypotheses is not too large and it is possible to implement the SOWWH test, this method will provide high power and low type I error. For large number of individual hypotheses which often is the case, the AU test is recommended for the constraint test, where we can easily increase the number of bootstraps in the AU test to have more accurate $P$-values. The FDR controlling procedure is a safe choice, which works well along with the SOWWH and AU tests in any conditions. The Chi-square approximation method is powerful but the zero $P$-value issue must be dealt with. The effect on the overall statement of using the midpoints between 0 and smallest possible $P$-values to replace zero $P$-values is not clear and it is difficult to avoid zero $P$-values without dramatically increasing the number of bootstrap replicates.
4.6 Real Data Analysis

We implemented the procedure of the constraint test on the data set with 46 taxa and 443 amino acid positions in Andersson and Roger (2002). The ML tree of the 46 taxa based on 6-phosphogluconate dehydrogenase (gnd) - the second enzyme in the oxidative pentose phosphate pathway - is given in Figure 4.4. From their analysis, we know that with the exception of five eukaryotic sequences, Dictyostelium discoideum (Dd), Trypanosoma brucei (Tb), Leishmania major (Lm), Trichomonas vaginalis (Tv) and Giardia lamblia (Gl), the eukaryotes group are divided into two distinct splits: "plant+protist" (PP) split with 10 taxa and "animal+fungal" (AF) split with 8 taxa in the ML tree. The question is whether these five exceptions are accidentally outside of the eukaryotes groups due to error.

Except for these 5 taxa, there are 41 taxa remaining and are divided into three groups: PP split with 10 taxa, AF split with 8 taxa and $46 - 5 - 8 - 10 = 23$ eubacteriums. To address the above question, 10 tests are formed, which are that Dd+PP, Dd+AF, Tb+PP, Tb+AF, Lm+PP, Lm+AF, Tv+PP, Tv+AF, Gl+PP or Gl+AF are in the same split. Note that these hypotheses are slightly different from the null hypothesis (1) in Section 2. Each of these hypotheses only tests that one taxon is attached to a group of taxa, which is different from testing all of them are in the same split. We illustrate this point using the first test as an example.

The overall hypothesis of Dd and PP split is

$$H_0 : \text{Dd is in the split of PP}$$

In other words, Dd is closer to the taxa within the PP split than the taxa in the residual split under $H_0$. An assumption we have to make in the above test is that PP split is true relative to the residual split, which means none of the taxa have been misplaced between these two splits. Without this restriction, we can't draw the conclusion of the overall hypothesis if it is rejected. The rejection may be caused by misplacement of Dd itself, or by any other misplaced taxa. To decrease the computation time by using smaller number of individual tests and also increase the accuracy, the taxa within the AF split are not used when tests involve the PP split. Only when both hypotheses (Dd+PP and Dd+AF) are accepted, we then test which split (PP or
AF) Dd belongs to. With AF split included, there are \( C_2^{11} \times C_2^{23+8} = 25575 \) individual 4-taxa tests; this number is decreased to \( C_2^{11} \times C_2^{23} = 13915 \) with AF split excluded. Since \( H_0 \) only tests the relation of Dd with PP split and the residual split, thus it suffices to test the 4 taxa trees which are formed by Dd, one taxon from PP split and 2 taxa from the residual split. This further decreases the total number of individual tests to \( C_1^{10} \times C_2^{23} = 2530 \). Our gain in accuracy of the test is substantial: the \( P \)-values should be accurate to \( 0.05/2530 = 2 \times 10^{-5} \) in order to compare to \( \alpha/m \) in the multiple comparison methods, which means that we at least need roughly 50,000 replicates for each individual test in the SSOWH and AU tests. The tests tend to reject more when the number of bootstrap is not large enough.

However, even after the above treatment, the SSOWH test is still not executable in this case. The process will be extremely time consuming even for only one test instead of 2530. Thus the AU test is the only choice we have for this example. In order to increase the accuracy of the AU test, instead of using the default setting of CONSEL, we used the setting with the scales in bootstrap from 0.5 to 1.5 with an increment of 0.1 and 100,000 bootstraps for each scale.

Based on FDR controlling procedure, the number of rejections of the individual tests for these 10 constraint tests are listed in Table 4.4. The number of rejections for the first row in Table 4.4 is out of 2530, for the second row is out of 2024. Note that we reject the overall hypothesis if there is at least one individual hypothesis rejected. Based on the results, we can conclude that Dd is in the AF split, and Tb and Lm are in the PP split. Both Tv and Gl are rejected to be in either split of PP and AF with more rejection times for the AF split. The number of rejections of Gl+PP and Tv+PP are not large, only 2 and 18 respectively. Since all the rejected tests have zero \( P \)-values, we enlarged the number of bootstraps to 1,000,000 for these two cases but the results don’t change. Note the results of the 10 constraints in Chi-square approximation method are consistent with the FDR controlling procedure after replacing the zero \( P \)-values by \( 1/(2*1,000,000) = 5 \times 10^{-7} \). There are two possible reasons if the two hypotheses are falsely rejected. One is that the output \( P \)-values in CONSEL only have three decimals. It is possible that the \( P \)-values are treated as 0 if they are less than 0.0005. The other is that even if the method controls FWER at
significance level 0.05, there are still some chances to get false rejections. Note that the number of bootstraps in the AU test may lead to serious problem if it is not large enough. With the default setting of CONSEL, the constraints Tb+PP and Lm+PP are rejected because of a single rejected test with $P$-value being zero.

<table>
<thead>
<tr>
<th>constraints</th>
<th>Dd+PP</th>
<th>Tb+PP</th>
<th>Lm+PP</th>
<th>Tv+PP</th>
<th>Gl+PP</th>
</tr>
</thead>
<tbody>
<tr>
<td>number of rejected $H_{0i}$</td>
<td>93</td>
<td>0</td>
<td>0</td>
<td>18</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>constraints</th>
<th>Dd+AF</th>
<th>Tb+AF</th>
<th>Lm+AF</th>
<th>Tv+AF</th>
<th>Gl+AF</th>
</tr>
</thead>
<tbody>
<tr>
<td>number of rejected $H_{0i}$</td>
<td>0</td>
<td>405</td>
<td>31</td>
<td>161</td>
<td>608</td>
</tr>
</tbody>
</table>

Table 4.4: The number of rejections of the individual hypotheses for the 10 constraints based on combination of the AU test and the FDR controlling procedure.

If Gl+PP is a false constraint, the two rejected topology (\((\text{Gl}, \text{Porphyra yezeoensis} 1), (\text{Shewanella putrefaciens}, \text{Chlamydia pneumoniae})\) and (\((\text{Gl}, \text{Porphyra yezeoensis} 1), (\text{Shewanella putrefaciens}, \text{Synechococcus sp. WH8102})\) can tell us some information about the true structure. The details of the rejected trees in Tv+PP test won’t be given here. We believe that even if these two constraints are not true, these two species must still be very close to the PP split.

### 4.7 Conclusion

We developed a test of splits based on ordinary tree tests. It has two features. The first is that only 4-taxon trees are tested where the assumption of the star tree as the boundary is not invalid anymore and the requirement of long sequence is not critical. The second is the large number of individual hypotheses, which requires fast computation and accurate $P$-values. Although the AU test is a conservative choice as an ordinary test, it fits well in these two aspects and provides strong power with reasonable rate of type I error. By applying our method on the data set of the example in Andersson and Roger (2002), a clear picture about the misplacement of taxa in the ML tree was reached.

Through simulation studies, we found the AU and SSOWH tree test methods combined with either of FDR or Chi-square approximation are performing well. The computation load of AU test is much smaller than that of SSOWH test. To avoid
dealing with the independence assumption of $P$-values and zero $P$-value issues in the Chi-square approximation method, we prefer FDR method more than the Chi-square approximation method even though the type I error and power of these two multiple test methods are comparable.
Figure 4.4: The ML tree of 46 taxa.
Chapter 5

Pattern Classification of Phylogeny Signals

We discuss clustering analysis based on the phylogenetic signals of genes in this chapter. Though it is natural and easy to compare the ML trees of the genes for clustering, it is not suitable for some situations, such as when the ML tree is not consistent with the true tree and several trees have similar strong signals because of the proximate star topologies.

Our goal is to find an appropriate method to cluster genes based on their phylogenetic signals. An appropriate method should provide a good clustering result and a reasonable estimate of the number of clusters. We propose two methods especially designed for phylogeny clustering: minimum entropy clustering (MEC) and the two-stage clustering (TSC). The first method solves problems from the entropy point of view, and the second one is composed of two consecutive procedures, each controlled by different criterion.

The rest of the chapter is organized as below. How to express phylogenetic signals is our first question, which is discussed in section 5.1. We choose likelihood weights as the inputs for the clustering methods and their properties as phylogenetic signals are discussed. The $P$-values from the test methods as phylogenetic signal inputs are included for comparison. Section 5.2 introduces distance measures which are the essential part of clustering methods, such as within-cluster and between-cluster distances. As a tool for estimating the number of clusters and evaluating criteria, the gap statistic (Tibshirani, Walther and Hastie 2001), based on the within-cluster distances is also reviewed. The two proposed clustering methods are presented in detail in section 5.3. The simulation analysis is introduced in section 5.4 and the evaluation of the proposed methods based on two simulated data sets is included in section 5.5. At first the performances of likelihood weights and $P$-values as inputs in the hierarchical clustering method are compared, which strongly suggest that likelihood
weights are better choices with smaller clustering errors. Then the implementations of the two proposed methods are illustrated with likelihood weights as inputs. The MEC method provides reasonable estimated number of clusters and small clustering errors for the two examples. Though TSC gives the same clustering result as the hierarchical clustering method with the same agglomeration criterion, it doesn’t mean the former method is only as good as the latter one. It has potential to perform better in some difficult situations or large data sets because of the separation control in the first stage. After the summary and discussion of their performance in section 5.6, the two new methods are applied to a real data set with 510 genes in section 5.7. Note that the clustering results are only demonstrated by heatmap without biological explanations.

5.1 Phylogenetic Signals

The evolutionary history of different genes is represented by their phylogenetic signals. Since the ML tree may not be consistent with the true evolutionary process, we don’t only consider the ML trees but the signals over all the phylogenies. Let a $n \times m$ matrix, $X$, denote the input data composed of signals, where $n$ is the number of genes and $m$ the number of candidate trees. A similar pattern in rows indicates the same evolutionary process. Here we consider likelihood weights and $P$-values from test methods as phylogenetic signals for clustering methods.

5.1.1 Likelihood Weight (LW)

As in chapter 3, a likelihood weight (LW) is a proportion of the likelihood of a tree over the sum of the likelihoods of all possible trees (see formula 3.6). The elements in $X$ can be $LW_{ij}$, denoting the likelihood weight of $\tau_j$ of gene $i$, where the sum of each row is 1. We don’t use likelihood directly as the input because it depends on the sequence length. Different genes have different sequence lengths and long sequence lengths tend to have smaller likelihood than short ones. It is possible that the genes are clustered together just because of their similar sequence lengths if the likelihoods are used as input.

If we suppose a uniform prior on $m$ trees and replace the probabilities of trees
being the true tree by their likelihoods, the LW can be considered as a Bayesian "posterior". If we suppose that the true tree is one of the candidates, the LWs can be understood as the probabilities in a multinomial distribution. But it is not a pure Bayesian posterior because the likelihoods depend on the parameters and \( m \) may be a number much smaller than the number of possible unrooted trees for large number of taxa.

5.1.2 \( P \)-values

The \( P \)-values measure the possibilities of the candidate trees as the true tree based on the observations from hypothesis testing point of view. As the input data, \( P_{ij} \) denotes the \( P \)-value of testing \( \tau_j \) as the true tree for gene \( i \). The larger the \( P \)-value is, the more possible that the candidate tree is true tree.

However the \( P \)-values are not fixed properties of topologies. They can be extremely different based on which test is used. We use the \( P \)-values from the AU test as examples in this chapter. Note that how to choose the appropriate test method to calculate \( P \)-values is still an unsolved problem (Shi et al. 2005).

5.2 Distance Measures and Gap Statistic

Before we introduce any clustering method, distance measures must be brought into consideration, which is the essential part of many distance-based clustering methods. The gap statistic which can be used to estimate the number of clusters is also based on within-cluster distances. In the following, we will give the detailed descriptions.

5.2.1 Distance Measures in Clustering Methods

We can divide distances into three types in gene clustering problems: distance between genes, distance within clusters and between clusters.

Let \( d(i, i') \) denote the distance between genes \( i \) and \( i' \). Euclidean distance (ED) is the most common choice for \( d(i, i') \), which examines the root of squared differences between coordinates of a pair of objects. As a natural distance measurement for the
$P$-values as inputs, it is defined as

$$d(i, i') = ED(i, i') = \sqrt{\sum_{j=1}^{m} (P_{ij} - P_{i'j})^2}$$

However for the LWs, Euclidean distance ignores the fact that the sum of each row in $X$ (LWs matrix) is 1. It is more reasonable to measure $d(i, i')$ by Kullback-Leibler (KL) divergence (Kullback and Leibler 1951) which is defined as

$$d(i, i') = KL(i||i') = \sum_{j=1}^{m} (LW_{ij} * \log(\frac{LW_{ij}}{LW_{i'j}}))$$

for the multinomial probability distributions of gene $i$ and $i'$. $KL(i||i')$ measures the expected amount of the information that a probability distribution of gene $i'$ is different from the probability distribution of gene $i$. The smaller the KL divergence is, the closer the two distributions are. Note that if a probability in the distribution of gene $i'$ is 0 which corresponds to a degenerated multinomial distribution for gene $i'$ but the corresponding probability in the distribution of gene $i$ is not 0, the KL divergence is defined as infinity so that the two distributions have the largest divergence. This causes problems in situations where two distributions are indeed very similar with one distribution degenerated while the other close to be degenerated. Further more, the KL divergence is not a real distance due to the fact that $KL(i||i') \neq KL(i'||i)$. It may become a nuisance in some applications due to its lack of symmetry. In order to avoid the degenerated distribution problem and have a symmetric distance measure that can be easily evaluated, the capacitory discrimination (CD) is considered. It is defined as

$$d(i, i') = CD(i, i') = KL(i||c) + KL(i'||c)$$

where $c$ is the mean of the two distributions and is still a probability distribution. The capacitory discrimination is symmetric and behaves like the KL divergence for small distances.

Based on the gene-gene distance $d(i, i')$, the within-cluster (WC) and between-cluster (BC) distances can be calculated if the clusters are known. Suppose there are $k$ clusters $C_1, ..., C_k$, with $C_r$ denoting the index of observations in cluster $r$, and $n_r$ denoting the size of $C_r$. The definitions of the two distances can be divided into two
categories. One considers the pairwise distances between genes referred to as "pairs", and the other involves the centers of the clusters, referred to as "centers".

Generally, the WC distance of \( k \) clusters can be defined as

\[
WC = \sum_{r=1}^{k} \frac{1}{2n_r} \sum_{i,i' \in C_r} d(i, i') \quad \text{(pairs)} \tag{5.1}
\]

or

\[
= \sum_{r=1}^{k} \sum_{i \in C_r} d(i, c_r) \quad \text{(centers)} \tag{5.2}
\]

where \( c_r \) is the center of \( C_r \), which may be the mean, the median or the centroid of all elements in this cluster. The denominator in Formula 5.1 makes the WC distance only dependent on the number of the clusters, not on the size of each cluster.

Similar to Formula 5.2, the BC distance involving centers can be defined as

\[
BC = \sum_{r=1}^{k-1} \sum_{r'=r+1}^{k} d(c_r, c_{r'}) \quad \text{(centers)}
\]

However, the definition of the BC distance between \( C_r \) and \( C_{r'} \) based on pairwise distances is not unique. The BC distance can be the distance of the closest pair of genes, or the most distinct pair of genes, or the average between all pairs of genes, where each pair is made up of one gene from each cluster, which are called single, complete or average linkages in hierarchical clustering and can be defined respectively as

\[
BC(C_r, C_{r'}) = \min \{d(i, i') : \text{where } i \in C_r \text{ and } i' \in C_{r'} \} \nonumber
\]

or

\[
\max \{d(i, i') : \text{where } i \in C_r \text{ and } i' \in C_{r'} \} \nonumber
\]

or

\[
\frac{1}{n_r n_{r'}} \left[ \sum_{i \in C_r, i' \in C_{r'}} d(i, i') \right] \quad \text{where } i \in C_r \text{ and } i' \in C_{r'}; \nonumber
\]

\( n_r \) and \( n_{r'} \) are the sizes of \( C_r \) and \( C_{r'} \) respectively.

In above, \( d(.) \) denotes the symmetric distance, Euclidean and CD. The WC distance is minimized as 0 when the number of clusters is equal to the number of genes, i.e. \( k = n \). The relationship between BC distance and number of clusters is not clear.

### 5.2.2 Gap Statistic

Many available clustering methods require prespecified information about the number of clusters which is not available in most situations. The gap statistic (Tibshirani,
Walther and Hastie 2001) provides a solution to get the optimal number of clusters based on the WC distance from the pairwise genes. It is based on the theory that the correct clustering result with the right number of clusters $K$ represents the true dispersion of the input data which minimizes the WC distance. However as we know, the WC distance decreases with increasing number of clusters no matter the clustering method is appropriate or not. Due to this fact, the WC distance is measured not only for a clustering result but also for its expectation under an appropriate null reference. The within-cluster distance with $k$ clusters, $WC_k$, is defined as following.

$$WC_k = \frac{1}{2n_r} \sum_{i,i' \in C_r} CD(i,i') \quad \text{for LWs} \quad (5.3)$$

$$= \frac{1}{2n_r} \sum_{i,i' \in C_r} ED^2(i,i') \quad \text{for P-values} \quad (5.4)$$

The gap statistic is defined as

$$\text{Gap}(k) = E[\log(WC^*_k)] - \log(WC_k) \quad (5.5)$$

where $WC^*_k$ denotes the reference value of $WC_k$ and $E$ denotes the expectation. Two techniques about how to generate replicate data sets for calculating its expectation were given in the original paper (Tibshirani, Walther and Hastie 2001). But neither is suitable for our cases since both P-values and the LWs must be positive and the sum of the LWs over one gene must be 1 in the replicates. In order to keep these properties, we calculate the reference $WC_k$ distances directly by randomly assigning genes to one of $k$ clusters. The average of $B$ reference values estimates the expectation of $\log(WC^*_k)$ with $sd(k)$ measuring its standard deviation. The standard error is

$$s_k = sd(k)\sqrt{1 + 1/B}$$

The estimate, $\hat{k}$, is chosen as the smallest $k$ such that $\text{Gap}(k) \geq \text{Gap}(k+1) - s_{k+1}$ which is so called ‘1-standard-error’ style of rule.

There are two motivations to consider $\log(WC_k)$ in gap statistic. One is that $\log(WC_k)$ is expected to decrease faster than its expected rate for $k \leq K$ and decrease more slowly for $k > K$, where $K$ is the number of well separated clusters. The other
is that $\log(WC_k)$ has an interpretation as a log-likelihood in the special case of a Gaussian mixture model (Scott and Simons, 1971).

5.3 Clustering Methods

In the following sections, we introduce two newly proposed methods: minimum entropy clustering (MEC) and two-stage clustering (TSC) which are specially designed for pattern classification of phylogeny signals.

The WC and BC distances are two regular criteria for evaluating the performance of a clustering method. A good method is expected to minimize the WC distance and maximize the BC distance. Because it is hard to satisfy both criteria at the same time, many clustering methods use only one of them. For example, to some extent, the hierarchical clustering method attempts to maximize the BC distance while the $k$-means method minimizes the WC distance. The TSC method combines the two criteria together by two stages: minimize the WC distance in the first stage and maximize the BC distance in the second stage based on the result from the first one. The MEC method is different to the ordinary distance-based clustering methods, in that it looks for clusters from a minimum entropy point of view. A gene is assigned to a cluster only when it supports the same tree as the genes in that cluster.

5.3.1 Minimum Entropy Clustering (MEC)

In information theory, entropy measures the uncertainty of a probability distribution. Shannon's entropy of a discrete random variable $Y$ with a probability function $P_Y(y)$ is defined as

$$\text{Entropy}(Y) = - \sum_y P_Y(y) \log P_Y(y) \tag{5.6}$$

which doesn’t depend on the actual values taken by the random variable $Y$, but only the probabilities. It is minimized when $P_Y(y) = 1$ for some $y$ and maximized if $Y$ is uniformly distributed. When $Y$ represents the likelihood weights of different trees, small entropy means that the gene provides strong signal about the true evolutionary history. The gene with the minimum entropy is the most informative one.
If two genes have the same evolutionary history, their probability distributions of the true tree should be similar and the entropy of the concatenated gene would be smaller than the entropy of either of them since there are more sites supporting their common trees. The MEC method is developed based on such reasoning. After the discussion of calculating entropy of the concatenated genes, we give the algorithm of this method.

**Entropy of Concatenated Genes**

Similar to the entropy of a single gene, the concatenated entropy is also calculated from a distribution with the concatenated LWs. We define the concatenated LW of $\tau_j$ for gene $i$ and $i'$ as

$$\frac{L_{ij} * L_{i'j}}{\sum_{l=1}^{m} (L_{il} * L_{lv_l})}$$

which is equivalent to

$$= \frac{L_{ij} * L_{i'j} / (\sum_{l} L_{il} * \sum_{l} L_{lv_l})}{\sum_{l} (L_{il} * L_{lv_l}) / (\sum_{l} L_{il} * \sum_{l} L_{lv_l})}$$

$$= \frac{L_{ij} * \sum_{l} L_{il} / \sum_{l} (L_{il} * L_{lv_l})}{\sum_{l} L_{il} * \sum_{l} L_{lv_l} / \sum_{l} (L_{il} * L_{lv_l})}$$

$$= \frac{LW_{ij} * LW_{i'j}}{\sum_{l} (LW_{il} * LW_{lv_l})}$$

where $m$ is the number of candidate trees. Then the concatenated entropy can be calculated from the new distribution by Formula (5.6). For example, if there are two genes with three possible tree structures, their probability distributions are

<table>
<thead>
<tr>
<th></th>
<th>$\tau_1$</th>
<th>$\tau_2$</th>
<th>$\tau_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>gene 1</td>
<td>0.8</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>gene 2</td>
<td>0.6</td>
<td>0.3</td>
<td>0.1</td>
</tr>
</tbody>
</table>

with the entropy values 0.64 and 0.9 respectively. Since gene 1 strongly supports $\tau_1$, it gives more information than gene 2. The concatenated LWs of gene 1 and 2 are

<table>
<thead>
<tr>
<th></th>
<th>$\tau_1$</th>
<th>$\tau_2$</th>
<th>$\tau_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>concatenated gene</td>
<td>0.52</td>
<td>0.52</td>
<td>0.52</td>
</tr>
<tr>
<td></td>
<td>0.92</td>
<td>0.06</td>
<td>0.02</td>
</tr>
</tbody>
</table>
where $0.52 = 0.8 \times 0.6 + 0.1 \times 0.3 + 0.1 \times 0.1$. So the concatenated entropy is

$$-(0.92 \times \log(0.92) + 0.06 \times \log(0.06) + 0.02 \times \log(0.02)) = 0.32$$

which is less than the entropy of gene 1 and 2. Though gene 2 is less informative than gene 1, its ML tree is still the same as gene 1, which indicates the same evolutionary history. The concatenated gene gives more support to the same ML tree, which is indicated by the decreased concatenated entropy. Note that the concatenated entropy is the same as the smaller one of the two genes if the LWs are uniformly distributed on all the possible tree topologies for the other gene.

It is not reasonable to concatenate amino acid sequences of two genes together to recalculate the likelihoods and LWs. During that process, the branch lengths and substitution model have to be reestimated. If the branch lengths are different which is very likely, model misspecification occurs even though the two genes have the same evolutionary history.

**Algorithm**

The core idea of this method is to put a gene in a cluster only if it shows support for the same tree as this cluster. The clustering procedure is composed by several similar consecutive steps. The procedure begins with the gene with minimum entropy among all the genes as the initial point which creates a cluster and is the center of this cluster. Then the genes are concatenated to the center of the cluster one by one to examine whether entropy can be decreased by concatenation. The order of genes for examination is obtained by their ascending CDs to the initial point. Note that this setting keeps the genes assigned to the cluster with similar evolutionary history to the initial gene. In other words, we can always tell the main evolutionary history of the genes in one cluster. The center is the mean of the distributions within this cluster, which needs to be recalculated after a new gene is assigned into this cluster. The step is over when none of the remaining gene can satisfy the criterion of reducing the overall entropy. If all the genes are clustered together, we believe that the dispersion among genes are not significant enough to separate them. In other words, there is only one single cluster. Otherwise, the gene with minimum entropy among the genes
outside of the first cluster is selected as the initial point for the second step and the
procedure repeats for the genes outside of the first cluster. The procedure continues
until all the genes are put into clusters. The number of the steps is decided by the
input data itself and indicates the number of clusters.

The implementation of this algorithm for gene clustering is described as below.

**Input:** The input data set $X$ containing $n$ genes ($X$ can be only composed of LWs).

**Output:** A set of clusters, the estimated number of clusters is given automatically.

1. Calculate the entropy for each gene in $X$ and use the gene with the minimum
   entropy as the initial point.
2. Order the genes by their CDs to the initial point.
3. Concatenate the current center of the cluster and the gene which is not in this
   cluster and closest to the initial point. If the entropy is not increased by the concate-
   nation, put the gene into this cluster and recalculate the center. Repeat step 3 until
   the concatenated entropy increases. This cluster is constructed completely.
4. Create a new $X$ with genes outside of the existing clusters and go back to step 1
to create another cluster. The whole process ends when all the genes are clustered.

Note that the procedure in step 3 doesn’t stop if the concatenated entropy stays
the same. The reason is that it is not crucial to put those genes with uninfor-
mative distributions into any cluster. Since the clusters are formed consecutively, the
uninformative genes tend to be in the clusters formed earlier.

There are two special properties of this algorithm. One is that unlike most avail-
able methods, this method provides the estimate of number of clusters along with the
cluster results. The other is that it doesn’t try to minimize the WC distance or BC
distance but minimize the concatenated entropy for each cluster.

### 5.3.2 Two-Stage Clustering (TSC)

Combining the two criteria (minimizing the WC distance and maximizing the BC
distance), is the main characteristic of the TSC method. The two criteria are realized
in two stages respectively. In the first stage, the genes are separated to several groups
which have the minimum WC distance based on a gene order given by singular value
decomposition (SVD). Then in the second stage, a pair of groups with smallest BC distance are combined to form one bigger cluster at each step until all the genes are joined together. The implementation of the second stage is similar to hierarchical clustering. While the hierarchical clustering method begins with single genes, TSC begins with the groups obtained from the first stage. The extra step (first stage) makes it less likely to put genes with similar signals into different clusters.

In the following, the presentation for finding the gene order is given first. Then the stop criterion of the first stage is discussed. Without controlling, \( n \) groups can be created with only one gene in each group, which makes this method equivalent to the hierarchical clustering method. We adapt the idea of gap statistic to control the first stage. Finally the algorithm of this method is given.

**SVD**

The singular value decomposition (SVD) of a rectangular data matrix \( X \) can be used to understand the structure of the data and give insight into the relationships of the row (genes) and column (trees) factors. The SVD of the matrix is

\[
X = A\Lambda B^T = \sum_{i=1}^{m} \Lambda_i a_i b_i^T
\]

where \( A = (a_1, \ldots, a_m) \) is the \( n \times m \) column-orthogonal matrix with \( a_i \) as the left eigenvectors, \( \Lambda = \text{diag}(\lambda_1, \ldots, \lambda_m) \) the \( m \times m \) diagonal matrix of eigenvalues, and \( B = (b_1, \ldots, b_m) \) the \( m \times m \) orthogonal matrix with \( b_i \) as the right eigenvector. The first column of the left eigenvectors, \( a_1 \) helps us to find the order of the genes, which tells the projection of the original observations in the main direction given by the first right eigenvector \( b_1 \). The gene order is decided by their corresponding values in \( a_1 \). The genes with similar distributions are close to each other based on the order of the elements in \( a_1 \) and they are separated to minimize the within-cluster distance. Since the initial order of all the genes are given only based on main directions \( b_1 \), the genes in each group after each step of separation are reordered by \( a_1 \) which is obtained from SVD on this smaller group.
Stop Criterion of the First Stage

The first stage is for separating genes with different signals. In each step, we create one more group based on the separation from the previous step, which is realized by two moves. First, we divide each of the \( i \) groups from the last step into two groups according to the order given by SVD and the cuts are chosen to minimize the WC distances of these groups respectively. Secondly, we pick the group and the cut which reduces the WC distance the most to create \( i + 1 \) groups. By these two steps, we find the separation to minimize the overall WC distance from \( i \) to \( i + 1 \) groups.

However there is no criterion to tell when to stop the separation. In other words, how to evaluate the level of the genes similarity within the groups is not known. Here we adapt the principle of gap statistic to control this stage by defining an adapted gap statistic (AGap) as

\[
AGap_k = E(WC_k^*) - WC_k
\]  

(5.7)

where \( WC_k \) is defined as Formula 5.3. Let \( \max(\text{AGap}) \) be the maximum statistic and \( s_{\text{max}} \) be its corresponding standard error. The number of groups from the first stage is decided as

\[
L = \max \{ l \mid AGap(l) > \max(\text{AGap}) - s_{\text{max}} \}
\]

which is the largest number of groups which has gap statistic within one standard error of the maximum adapted gap statistic.

In contrast to the original design of the gap statistic to prevent too many clusters, our main problem here is to avoid the insufficient separation. Though an excess of separation may divide the similar genes into different groups, the genes still have chances to be combined back in the second stage. The only possible cost is that the performance of the TSC method may be similar to the hierarchical clustering. However an insufficient separation causes much more serious problems, i.e. the genes with different signals are not separated into different clusters. That’s why we consider the maximum value of the adapted gap statistic, which makes estimate of the groups larger than the original design of the gap statistic. However replacing \( \text{AGap} \) by \( \text{Gap} \) (Formula 5.7) makes the number of groups, \( L \), too large, close to number of genes \( n \). Generally \( E(WC_k^*) \) is larger than 1 and decreases very slowly, while \( WC_k \) decreases.
Figure 5.1: Comparison of the adapted gap statistic and the gap statistic on estimating the number of groups.

much faster with increasing $k$, which may be less than 1. Their logarithm difference keeps increasing in this situation, which leads to large number of groups. Figure 5.1 is an example to illustrate this statement based on a data set with 100 genes. In this figure, the x-axis is the number of groups from 0 to 48, and the y-axis is adapted gap statistic (defined in Formula 5.7) in the left panel and gap statistic (defined in Formula 5.5) in the right panel. While the maximum adapted gap statistic is given with 16 groups, the maximum value of gap statistic is not achieved even with 48 groups. In other words, if we use the gap statistic to estimate the number of groups, $L$, the estimate will be larger than 50, which means an average there are less than 2 genes in each group. Under this situation, the first stage may not add any useful information to the TSC method compared to hierarchical clustering.

Algorithm

Overall, this algorithm can be summarized as having the following two parts. 1) Get a series of groups and estimate the number of groups using the adapted version of the gap statistic; 2) Combine the groups into successively larger clusters one by one until all genes are merged together.

The details of this algorithm is given below.

**Input:** A data set $X$ containing $n$ genes.
**Output:** A series set of clusters, at most $L$ possible clusters, where $L$ is the number of groups obtained from the first stage.

1. Find the order of genes in $X$ by SVD and cut them into two groups, which minimizes the WC distance of two groups. Since currently all the genes are in one large group, the cut of this group is the one to minimize the overall WC distances.

2. Find the order of genes by SVD in each group and cut each group into two. The cut in each group is chosen to minimize the WC distance of two parts of this group. In order to avoid excess separations, we only keep the one out of all cuts which minimizes the overall WC distance of $X$. Repeat step 2 until there are $n/2$ ($n$ is even) or $(n+1)/2$ ($n$ is odd) groups.

3. Apply the adapted gap statistic to estimate number of groups $L$ and use these groups as the primary elements of the second stage.

4. Calculate the BC distances between groups and merge the two with the smallest distance together. The procedure stops when all the genes are merged into one large cluster.

5. Estimate the number of clusters by gap statistic which is less than $L$.

Theoretically, step 2 should stop after $n$ groups are generated, which makes each gene in a single group. However based on our experience, it is enough to have $n/2$ (or $(n+1)/2$) groups to find the optimal number of groups by adapted gap statistic. The groups generated from the first stage are nested. Note that we have to estimate the number of clusters if there is no prior information available. Gap statistic is one of the choices for estimation.

5.4 Simulation Analysis

The illustration of the two proposed methods is based on two simulated data sets. The software used and the underlying models for simulations are presented first. Then the information about how to obtain the cluster results and evaluate their performance are given. Heatmap is introduced in order to give a visual demonstration of the clustering results. At last the simulated data sets are given and demonstrated by heatmaps.
5.4.1 Design of the Simulations

In order to evaluate the clustering methods, we simulated amino acid data from 5-taxon model trees using \textit{PSeq-Gen} (Grassly, Adachi and Rambaut 1997) so that the true partitions are known. \textit{PSeq-Gen} simulates the evolution of protein sequences under substitution models and along evolutionary trees, similar to \textit{Seq-Gen}.

Fixing JTT (Jones et al. 1992) as the substitution model, the parameter in the gamma distribution used for rate heterogeneity was selected from 0.57 to 2.86, in increments of 0.03. The sequence lengths are from 100 to 1553, which is the range of the sequence lengths in our real data with 510 genes. There are 5 external and 2 internal branches in each 5-taxon tree. Even for the trees with the same structures, the branch length were set up differently. Most internal and all external branch lengths were randomly selected from 0.005 to 0.5, in increments of 0.005. In order to increase the difficulty, we also set the range of one internal branch lengths from 0.005 to 0.05, which makes it less likely to have a strong signal about the true tree. The genes simulated with short internal branch lengths are called \textbf{difficult genes}.

All the fifteen 5-taxon unrooted trees are considered as the candidate trees to construct the input data (m=15). The tree topologies are labeled in the same order as in Figure 3.4 and are referred to by the corresponding index for convenience.

\begin{table}[h!]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
 & Group & 1 & 2 & 3 & 4 & 5 \\
\hline
\textbf{Example 1} & Gene & 1-20 & 21-40 & 41-60 & 61-80 & 81-100 \\
 & ML tree & 5th & 9th & 10th & 1st & 12th \\
\hline
\textbf{Example 2} & Gene & 1-70 & 71-73 & 74-76 & 77-80 & 81-100 \\
 & ML tree & 5th & 9th & 12th & star tree & 10th \\
\hline
\end{tabular}
\caption{The information of the underlying model trees in the two examples.}
\end{table}

Two examples are simulated to compare the performance of the clustering methods. Each of them comprises 100 genes within 5 underlying trees, one denoting the balanced case with the same number of genes in each group, and the other the unbalanced one including one extremely large group and few outliers. More details are given in Table 5.1.

In the first example, there are 20 genes in each group. It means that gene 1-20, 21-40, 41-60, 61-80 and 81-100 are in the same groups respectively. Five genes in each
group (16-20, 36-40, 56-60, 76-80, 96-100) are difficult genes since their underlying tree models are close to a star tree, which make it hard to cluster the genes correctly. In the second example, gene 1-70 are in one big group, 71-73, 74-76, 77-80 in three small ones and 81-100 in a medium size one. Gene 51-70 and 96-100 are difficult genes. Gene 77-80 are simulated from a star tree with two zero internal branch lengths, which indicates that they can be put into any cluster.

5.4.2 Analysis

As the input data, the $P$-values of the AU test were obtained by CONSEL with the site log-likelihoods from PAML and the LWs by R, version 2.1.1, after the likelihoods were given by PAML. Note that the gamma distribution was assumed for rate variation across sites in PAML. The MEC and TSC methods were only implemented on the LWs. Using ED or CD to measure the similarity of the genes, the hierarchical clustering method was applied on both kinds of inputs. There are two purposes for its implementation. One is to show the advantage of LWs over $P$-values as inputs, and the other is to check the effect of the first stage of the TSC method. For the second purpose, we used the same criterion for the hierarchical clustering and the second stage of the TSC method. All the methods were implemented in R.

<table>
<thead>
<tr>
<th>method</th>
<th>input</th>
<th>distance</th>
<th>center</th>
<th>criterion</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEC</td>
<td>LW</td>
<td>CD</td>
<td>mean</td>
<td>entropy</td>
</tr>
<tr>
<td>TSC</td>
<td>LW</td>
<td>CD</td>
<td>mean</td>
<td>the WCD with centers for separation the complete linkage for aggregation</td>
</tr>
<tr>
<td>Hierar</td>
<td>LW</td>
<td>CD</td>
<td>NA</td>
<td>complete linkage</td>
</tr>
<tr>
<td></td>
<td>P-value</td>
<td>ED</td>
<td>NA</td>
<td>complete linkage</td>
</tr>
</tbody>
</table>

Table 5.2: The summary of the options in the clustering methods.

Table 5.2 summarizes the features in the hierarchical clustering, MEC and TSC methods. We only present the results with complete linkage for the hierarchical clustering and TSC methods as an example. Though the WC, BC distances and centers of clusters can be defined in many ways, it is not the purpose of this chapter to find their best combination, but to evaluate their performances.
5.4.3 Heatmap Demonstration of All the Methods

Here the heatmap is used to provide a visual demonstration of the performance of the clustering methods, which is a data display method popular for genomic data. Basically a heatmap is a color display where the rows and the columns have been permuted to show interesting patterns. The basic idea of visualizing incongruencies through clustering of p-values was presented in Baptiste et al. (2005).

Heatmap in R

Figure 5.2: An example of the standard heatmap and the color palette of the corresponding LWs.

Figure 5.2 is an example of the typical heatmap based on the likelihood weights. The rows correspond to the genes and the columns correspond to the candidate trees. The likelihood weights are displayed, with dark spots for large value and light spots for small ones. The genes and candidate trees are clustered by hierarchical clustering based on their Euclidean distances. So the spots with similar color patterns across the rows were clustered together in the plot. The color and the corresponding LW are given in Figure 5.2 (b).
Heatmap in Chapter 5

While presenting the clustering results of different methods in this chapter by heatmap, the rows were not reordered by the hierarchical clustering, but by the examined clustering method. The candidate trees (the columns) kept the same order as before without reordering. A good clustering method should put the genes with the similar color patterns close to each other and in the same cluster.

5.4.4 Quantitative Evaluation of the Clustering Methods

In order to evaluate the clustering results quantitatively, we use the estimated number of clusters as one criterion. During the clustering procedure, we supposed that there was no prior information about the number of clusters. The closer the estimate to the truth, the better the clustering method is. When using gap statistic to estimate the number of clusters for TSC, the standard error of the expectation value is based on 100 replicates.

Match error (ME) is another criterion for evaluation when the true partition is known. It was proposed by Almudevar and Field (1999), which is defined as the minimum number of individuals that must be removed in order to make the partitions equal. This definition corresponds more closely to the intuitive notion of "the number of errors made" in the estimation of a partition. For example, the true and the estimated partitions are given in the following table. Because there is no prior information about the number of clusters, the estimated number of clusters is 4 instead of 3. The ME is 2 since the true and estimated partitions can be made equal by removing genes 10 and 9, and there is no way to remove less than two genes to make the two partitions equal.

<table>
<thead>
<tr>
<th>cluster</th>
<th>genes in cluster</th>
<th>estimated cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-4</td>
<td>1-4</td>
</tr>
<tr>
<td>2</td>
<td>5-9</td>
<td>5,6,7,8,10</td>
</tr>
<tr>
<td>3</td>
<td>10,11</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>9</td>
</tr>
</tbody>
</table>

Note that we have more chance to get small error when the estimated number of clusters is right.
5.5 Examples

In order to fully demonstrate the simulated data, we give the heatmaps (in Figure 5.3) of original LWs (a and b) and \( P \)-values from the AU test (c and d) for the two examples without reordering the genes and the candidate trees. The \( x \)-axis indicates the candidate trees 1-15 from left to right. The \( y \)-axis indicates the genes 1-100 from bottom to top. The dark spots show large LWs or \( P \)-values, while the light ones for small ones. For the LWs, since the dark spot indicates the LW around 0.9, it is only possible to have one dark spot for one gene. The rows (or genes) with one dark spot means they have strong signal about the true evolutionary history. The genes with equally light color on 15 trees don’t have the strong signal, which are difficult for clustering methods to assign them correctly. Ideally, the genes in the same cluster should have dark spots on the same column, indicating the same ML tree. The heatmap of the \( P \)-values are different. It is possible that there are more than one dark spot for a gene since there may be more than one tree with very high \( P \)-value.

The genes are in 5 groups based on the underlying models. The heatmaps show that there are contradictions between the phylogenetic signals and the underlying models. For example, though gene 97 is simulated from underlying tree 10, there is a dark spot on the fourth column (at the top of plot (b) in Figure 5.3), representing 0.935 as the LW. It is not a surprise that any clustering method can’t assign it to the right cluster. And from the heatmap, it seems that the information of the underlying model is consistent more with the signal of the LWs than \( P \)-values.

This section is composed of three parts. The first one is to evaluate the two inputs: LW and \( P \)-value, by comparing their clustering results from hierarchical clustering method. It is proved that LWs provide much smaller match errors than \( P \)-values. And that’s why we only consider LWs as inputs for the TSC method. (Only LWs are eligible for the MEC method). In the last two parts, we present the clustering results of MEC and TSC with heatmaps. The MEC method provides small match errors and reasonable number of clusters. The TSC method doesn’t outperform the hierarchical clustering in the two examples. But it has potentials to provide better results.
Figure 5.3: The heatmap of LWs (top) and P-values (bottom) of the two examples.
5.5.1 Comparison of the Input Data

The comparison of the input data is based on the clustering results of the hierarchical clustering. The results can be observed from the heatmaps in Figure 5.4 with the top two for the LWs and bottom two for the $P$-values. The order of the genes in heatmaps is decided by their agglomeration orders which is given automatically by the output of R. Note that the order is independent of number of clusters. The gene orders from the LWs make more sense than the one from the $P$-values. In other words, the genes having similar LW patterns are put together in the top two plots, e.g. the dark spots in the same columns are next to each other. But the pattern of the $P$-values is hard to tell.

Since it is not fair to make a conclusion only based on figures, the MEs of the two outputs are compared. We listed the MEs from 2 to 6 clusters in Table 5.3, in which the numbers prove that the LWs perform better than $P$-values by providing consistent smaller MEs.

<table>
<thead>
<tr>
<th></th>
<th>#clusters</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example 1</td>
<td>LW</td>
<td>60</td>
<td>45</td>
<td>27</td>
<td>21</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>$P$-value</td>
<td>79</td>
<td>79</td>
<td>62</td>
<td>61</td>
<td>56</td>
</tr>
<tr>
<td>Example 2</td>
<td>LW</td>
<td>15</td>
<td>13</td>
<td>20</td>
<td>24</td>
<td>36</td>
</tr>
<tr>
<td></td>
<td>$P$-value</td>
<td>35</td>
<td>36</td>
<td>36</td>
<td>36</td>
<td>36</td>
</tr>
</tbody>
</table>

Table 5.3: The comparison of MEs for LWs and $P$-values as input based on different number of clusters.

5.5.2 The MEC Method

The clustering results are summarized in Table 5.4. The top table is for the first example and bottom for the second. In each cluster, the genes are listed by the order that they are assigned to this cluster. And the heatmap demonstration of the clustering results are given in Figure 5.5 with left plot for the first and right for the second. The genes are presented by the order being assigned into clusters from bottom to top. Since the estimated numbers of clusters are given automatically, we use black lines to divide the genes into different clusters. The order of the genes being assigned into the same cluster indicates the strength of the phylogenetic signal.
Figure 5.4: The heatmaps of the hierarchical method for the LWs (top) and $P$-values (bottom).
<table>
<thead>
<tr>
<th>cluster</th>
<th>genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1-10-5-12-8-17-26-20-23-28-40-15-16-36-4-37-18-31-3-19-11</td>
</tr>
<tr>
<td>5</td>
<td>32-29-33-39-34-27-35-38-30-21</td>
</tr>
<tr>
<td>6</td>
<td>96-22</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>cluster</th>
<th>genes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-48-92-90-14-24-10</td>
</tr>
<tr>
<td>2</td>
<td>76-75-77-3-74-89</td>
</tr>
<tr>
<td>3</td>
<td>91-85-83-86-95-81-93-99-98-96-100-87-94-82</td>
</tr>
<tr>
<td>4</td>
<td>73-71</td>
</tr>
<tr>
<td>5</td>
<td>97</td>
</tr>
</tbody>
</table>

Table 5.4: The summary of the clustering results of MEC with the top table for the first example and bottom one for the second.

In the first example, the clustering procedure began with gene 42 from group 3. In plot 5.5(a), the first cluster at the bottom shows large LWs in the 10th candidate tree. Then the color fades in that column, which means that the signal becomes weaker and weaker. Then the second step began with gene 62 in group 4. The same procedure repeated, with the color in the first column changing from dark to light. Note that the genes with almost equal light colors between two groups have weak signals about their evolutionary history. This procedure ended with 6 clusters. In the first 5 clusters, though some of the genes are misclustered, most of the genes in each cluster have the same underlying model. Both genes in the sixth cluster have almost equal LWs on all 15 trees, which prevents them to be assigned to any of the previous 5 clusters. The ME is 22 in this example.

The procedure of the second example began with genes 30 in group 1, which was ended with 77 genes assigned into the first cluster. Gene 78-80 whose underlying model are full star trees were also assigned into this cluster, which is not a big issue to mix the genes simulated from a star tree with the other clusters. Then the procedure was repeated for another 4 times to create another 4 clusters. In other words, this method picks up the structure of the original data. The ME is 12 including the 4
Figure 5.5: The heatmaps of MEC with the left one for the first and the right for the second examples.

counts from genes with star topology.

5.5.3 The TSC Method

The demonstration of the clustering result of this method (Figure 5.6) is composed of three parts: the estimation of the number of groups in the first stage, the heatmap for the clustering result and the estimation of the number of clusters. The left column is for the first example and the right for the second one.

The scatter plots in the top show the adapted gap statistic to estimate the number of groups in the first stage. The dots give the values of Agap statistics (y-axis) for the corresponding number of groups (x-axis). The numbers of groups which give the largest Agap statistic were 16 and 11. The position of the line parallel to the x-axis tells the cut-off value obtained by subtracting the standard error of the maximum Agap statistic from the maximum Agap statistic. The largest number of groups with Agap statistic larger than the cut-off are 19 and 15 which are the estimated numbers
Figure 5.6: Demonstrations of adopted gap statistic, heatmap and gap statistic for TSC on the two simulated data sets.
of groups in the first stage. The corresponding separations were used as the primary elements in the second stage.

The heatmaps in the middle demonstrate the clustering results. In order to present the results, the genes are ordered by SVD within each group obtained from the first stage. Then the relationship between the groups is decided in terms of the order of the aggregation. It means that the two groups are next to each other in the heatmap if they are combined together in the step of the second stage.

The gap statistics for estimating the number of clusters are shown in the bottom plots. For a given number of clusters, there are two dots with the top one giving the value of gap statistic and the bottom one obtained by subtracting its standard deviation. The numbers of clusters are estimated as 4 for both examples, which are close to the truth. Note that only the partition with 4 clusters is given in the heatmaps by the solid lines. Comparing to the MEC method, the color of the dark spots doesn't fade constantly in a small area from bottom to top.

<table>
<thead>
<tr>
<th>cluster</th>
<th>genes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-40-3-19-31-1-5-8-10-12-17-21-30</td>
</tr>
<tr>
<td></td>
<td>-24-25-43-52-57-59-78-80-87-96</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>cluster</th>
<th>genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4-5-7-8-9-11-12-15-17-26-28-30-31-32-33-34-35-37-43-44-46-47-49</td>
</tr>
<tr>
<td>3</td>
<td>71-73</td>
</tr>
<tr>
<td>4</td>
<td>81-83-85-86-91-93-95-97-96-98-99-100-82-87-94</td>
</tr>
<tr>
<td>4</td>
<td>75-76</td>
</tr>
</tbody>
</table>

Table 5.5: The summary of the clustering results of TSC with the top table for the first example and bottom one for the second.

The clustering result with 4 clusters are given in Table 5.5. Still the top table is for the first example and bottom for the second. Though unlike the MEC method there is no order for clusters and genes within each cluster, we still order the cluster and genes based on the order in heatmaps from bottom to top in order to demonstrate
the results. With most of the underlying structures found correctly, gene 1-20 and 21-40 are merged together in the clustering result of the first example. It is partly due to the close ML trees in these two groups, which only need to exchange two external nodes to get the other tree structure. Most of the elements in the two large groups of the second example were picked up by this method. The four genes in the fourth group are assigned into the larger cluster with 81 genes. The results also indicate that this method pays attention to the distribution of the LWs. For example, while the MEC method assigned gene 74-76 with the other 3 genes in the one cluster in the second example, which have relatively higher LWs on at least one of trees 11, 12 and 13 than the other trees, the TSC method created a cluster with only gene 75 and 76 which have high LWs on all three trees 11-13. The MEs are 27 and 11 in the two examples, respectively.

In fact, this method provides the same clustering results as the hierarchical clustering method based on LWs for the two examples. It means that the first stage in the TSC method doesn’t make a difference on the final clustering result in the two examples.

5.6 Summary and Discussion

We compare the performance of three clustering methods, along with the two types of inputs. It is clear that the LW is a better choice than the $P$-values for clustering genes.

Overall, the MEC method performed the best for the two examples with small MEs and reasonable estimated number of clusters. Briefly speaking, the estimated number of clusters is decided by the number of the ML trees strongly supported by genes.

However observing the clustering process, we found that the performance of this method relies on the initial point of each step. In other words, the clustering result is sensitive to the genes with small entropy. Generally the clustering procedure in each step follows the descent order of the LWs of the ML tree of its initial point. The genes with blurred signals are more likely to be assigned to the clusters created first. It is also possible that the genes with clear but not strong signal supporting one ML
trees are assigned into a cluster with most of the genes supporting the other one, just because their LWs on the other tree are not extremely small, say 0.1 or 0.2. It is not clear whether it makes sense or not to put genes supporting different trees into the same cluster, especially when the LWs of the ML trees are greater than 0.5.

The TSC method didn't outperform the hierarchical clustering method in the two examples. But it doesn't mean that the first stage won't help to improve the performance. The separation in the first stage is not the reverse procedure of the aggregation because of the different criteria. It changes the order of group combinations comparing to single aggregation procedure. For example, if we use the separation result with 14 groups in the first example, we will get different clustering result with larger ME. This phenomenon indicates two things. First the first stage can change the clustering result. Secondly our adaptation of the gap statistic to estimate the number of groups in the first stage is feasible, which gave a good estimation to minimize the ME.

It is hard to say which one of the proposed methods is better since they solve the clustering problem from different points of view. The MEC method put more weight on the information about the ML trees and the TSC method focuses on the phylogeny signals of all trees. It is better to apply both methods in applications and find the common part of their results.

5.7 Real Data

The real data is comprised of 510 genes shared among four complete genome of the bacterial genus *Synechococcus* (NC005070, NC006576, NC007573, NC007516) and one complete genome of the bacterial genus *Prochlorococcus* (NC005071) from GenBank. The MEC method ended up with 12 clusters. The first three steps create three large clusters with size 81, 178 and 153. Most of the genes in the same cluster support the same ML tree, tree 2, 1 and 5. From plot (a) in Figure 5.7, we notice that there are some genes with darker color in other columns than the column of the ML tree of the cluster they are in. They may create clustering errors for the result.

In the TSC method, based on the 62 groups obtained from the first stage, the number of the clusters is estimated as 15 for the result of the TSC method. There
are two large clusters with size 112 and 140. They support candidate tree 1 and 5, respectively. The cluster which supports tree 2 has size 34, which is much smaller than 81. We can see that there is no color fading process in the bottom two clusters in plot (b) of Figure 5.7. The two clusters are clearly separated. In contrast with the MEC method by which there are clear phylogenetic signal in each cluster, the signal of the best tree is not always given by the TSC method. For example, there are no clear dark spots in one column in the middle of plot (b) in Figure 5.7. Note that the hierarchical clustering method does provide different result for this data.

For the case with unknown true partitions, such as the real data, the match error is not feasible to evaluate the performance of a clustering method. The WC and BC distances can be good measures under this circumstance. But they can only be compared for results with same number of clusters.

5.8 Conclusion

We proposed two clustering methods. The TSC method is developed from the hierarchical clustering method to avoid separating genes close to each other. There is some space to play different combinations of the WC and BC distance to get a good result. Since it is beyond the scope of this chapter, we didn’t focus on this problem. But it is an important topic for implementation on the real data. The MEC method solves clustering problems in terms of entropy. Clustering results based on phylogeny signals can be explained from many points of view. Our proposed methods provide more options for implementation and may give better results than others.
Figure 5.7: Demonstrations of the heatmaps of MSE and TSC, and adopted gap statistic and gap statistic for TSC on the real data
Chapter 6

Discussion

We discuss the contribution of the thesis and possible directions for future work.

6.1 Contribution of this thesis

The main contribution of this thesis are:

- Some discoveries to the methods for constructing confidence regions for phylogenetic trees (chapter 3):
  - The conservative performance of the SH and SSOWH test is due to the underlying assumption of the least favorable configuration.
  - The asymptotic chi-squares distribution of the test statistic in the GLS test requires longer sequence length with increasing number of taxa.
  - Being less sensitivity to model misspecification, nonparametric bootstrap is preferred to keep low type I errors.

- Two proposed methods: SDNB and SDPB, which improve the application of the bootstrap theorem on constructing the distributions of the test statistic under $H_0$ in phylogeny (chapter 3).

- Adoption of the test methods designed for fixed tree structure to hypotheses with unfixed ones by decomposing those unfixed structures to many 4-taxon trees with fixed structure (chapter 4). And the implementation of the proposed method on a real data provides biological reasonable results.

- Two proposed methods: MEC and TSC, which add more options for gene clustering (chapter 5).
6.2 Directions for further work

- Implementation of the confidence region methods including the SDNB and SDPB methods on real data which already has a good explanation based on biology knowledge to compare their performances.

- Development of a general diagnostic tool for the local structure of the phylogenetic topology when a large topology is constructed by any other methods based on the method in chapter 4. It is well known that searching of a large tree topology almost always results in a local optimum. Thus such a local diagnostic method will be very useful in complementing the tree search method. Furthermore, this method can also be used in a heuristic bottom-up search of the tree topology such that at each step we only combine two groups of taxa if the test is not rejected.

- Development of the MEC method to overcome its sensitivity to the genes with small entropy - a problem related to robustness. This problem, which is common for many clustering methods, can be alleviated by sampling subsets of the original data, by which a relationship matrix can be built to indicate the number of time that two genes are clustered together. We didn’t implement this idea in this thesis due to several difficulties.
  
  - How to sample the data, with or without replacement and what is the size of the subset.
  
  - How to decide the threshold in the relationship matrix to assign two genes in different clusters
  
  - How to decide the number of clusters since the result from each subset will be different

- Examination of the best combination of the two criterion in the TSC method, which is essential for the application in the real data. However different data sets may prefer different combinations, the clustering results have to be analyzed by biological knowledge to pick up the best combination for each data set.
• Application of the MSE and TSC methods on data with high-dimensional spaces. When the number of taxa is large, it is unfeasible to include all the possible trees as candidates. In other words, the trees with very small LWs have to be eliminated. However since it is likely that genes support many different ML trees, we have high-dimensional input data even only including the ML trees as candidates. One way to solve this problem is to put weights on trees, which is also interesting for low-dimensional data.

• Creation of R packages for all proposed methods.
Appendix A

Tree Structures in Figure 3.2

The trees are in NEXUS format, and can be viewed in PAUP*.

10-taxon tree
(1: 0.013830, 2: 0.015832, (3: 0.029746, ((4: 0.017846, 5: 0.014022): 0.032036,
6: 0.047509, (7: 0.057006, (8: 0.026139, (9: 0.021788, 10: 0.032609): 0.001234):
0.038943): 0.001426): 0.006973): 0.004049): 0.017251);

15-taxon tree
(1: 0.013935, 2: 0.015366, (3: 0.033192, ((4: 0.023817, 5: 0.015218): 0.038405,
6: 0.042860, (7: 0.056182, (((8: 0.023098, 9: 0.024827): 0.000432, 10: 0.023845):
0.015672, (((11: 0.005724, 12: 0.006404): 0.001903, 13: 0.005464): 0.028134, 14:
0.037194): 0.001776, 15: 0.047241): 0.023468): 0.015238): 0.001294): 0.004894):
0.004605): 0.015000);

20-taxon tree
(1: 0.013823, 2: 0.015819, (3: 0.029629, ((4: 0.017817, 5: 0.014036): 0.031956,
6: 0.047519, (7: 0.057420, ((8: 0.026198, (9: 0.022117, 10: 0.032245): 0.001317):
0.015635, ((((11: 0.004742, 12: 0.007728): 0.000829, 13: 0.005867): 0.035062,
(14: 0.043034, 15: 0.046981): 0.001445): 0.017608, 16: 0.059090): 0.001133, ((17:
0.072962, 18: 0.051127): 0.011258, (19: 0.020931, 20: 0.013127): 0.029135): 0.013228):
0.000914): 0.022642): 0.001226): 0.006938): 0.003981): 0.017363);
Bibliography


