Neural, Parallel & Scientific Computations 15 (2007) 547-560

DISTANCE BASED METHODS IN PHYLOGENETIC TREE CONSTRUCTION

CHUANG PENG[†]
DEPARTMENT OF MATHEMATICS
MOREHOUSE COLLEGE
ATLANTA, GA 30314

ABSTRACT. One of the most fundamental aspects of bioinformatics in understanding sequence evolution and relationships is molecular phylogenetics, in which the evolutionary histories of living organisms are represented by finite directed (weighted) graphs, in particular, directed (weighted) trees. There are basically two types of phylogenetic methods, distance based methods and character based methods. Distance based methods include two clutering based algorithms, UPGMA, NJ, and two optimality based algorithms, Fitch-Margoliash and minimum evolution [7]. This paper focuses on distance based methods. The paper starts with some preliminary knowledge and definitions in the area, including finite directed graphs, directed trees and matrices. It discusses the verification of the metric property of distance matrices, including detections of errors if a distance matrix fails to satisfy the metric property, and then provides an algorithm in modifying the distance matrix to satisfy the metric property. The second part of the paper is a brief survey based on the excerpts from the references, on various frequently used distance based phylogenetic tree construction methods, both cluster-based and optimality base methods, including UPGMA, Neighbor Joining, Fitch-Margoliash, and Minimum Evolution methods. Also, it discusses the assessment of the phylogenetic trees and some analysis of the algorithms.

§1. Introduction

Phylogenetic is the study of the evolutionary histories of living organisms, and represent the evolutionary divergences by finite directed (weighted) graphs, or directed (weighted) trees, known as *phylogeny*. Based on molecular sequences, phylogenetic trees can be built to reconstruct the evolutionary tree of species involved. In particular, the representation derived from genes or protein sequences is known as *gene phylogeny*, while the representation of the evolutionary path of the species are often referred as *species phylogeny*. A gene phylogeny is, to some extend, a local desciption. It only describes the evolution of a particular gene or encoded protein, and this sequence could evolve much more or less differently than other genes in the genome, or it may have completely different evolutionary history from the rest of the genome due to horizontal gene transfers. Therefore, the evolution of a particular gene only provides a local picture, not necessarily reflect the *global* evolutionary picture of the

[†]This short study note is based on a bioinformatic research project at Stanford University and references listed, and has been posted as class materials on Douglus Brutlag's *Computational Molecular Biology* class webpage at Stanford University. Partially sponsored by Department of Education MSEIP grant P120A060097 and Morehouse College.

1061-5369 $15.00 © Dynamic Publishers, Inc.

species. We can only hope that we could assemble the jigsaw puzzle pieces with a wide selection of gene family to give an overall assessment of the species evolution.

While in general the topology in phylogenetic trees represents the relationships between the taxa, assigning scales to edges in the trees could provide extra information on the amount of evolution divergence as well as the time of the divergence. The phylogenetic trees with the scaled edges are called *phylograms*, while the non-scaled phylogenetic trees are called *cladograms.* Purely for the sake of computers data processing, some special formats were artificially created, such as *Newick* format. From biological point of view, the building of phylogenetic trees can be roughly divided into the following steps .

Choose molecular marker. In building molecular phylogenetic trees, either nucleotide or protein sequence data can be used, but the outcomes from the choice could be quite different. The rule of thumb is to select nucleotide sequences when some very closely related organisms are studied because they tend to evolve more rapidly than proteins; and to select protein sequence (or slowly evolving nucleotide sequences) when more widely divergent groups of organisms are studied.

Perform sequence alignment. The sequence alignment establishes positional correspondence in evolution, and aligned positions are assumed to be genealogically related. Though there have been numerous so-called "state-of-art" alignment programs available and many times they do help, manual editing is often crucial in the quality of alignment, and yet there is no firm or clearly defining rule on how to modify a sequence alignment.

Choose a model of evolution. One quantitate measure of divergence between two sequences is the number of substitutions in an alignment. However, this measure can somehow be misleading in representing the true evolution. Not only the nucleotide may actually undergone several intermediate steps of changes behind a single mutation in the sequence, but also an observed identical nucleotide may hide parallel mutations of both sequences. This is known as *homoplasy.* The statistical models used to correct homoplasy are called *evolutioinary models.* For constructing DNA phylogenies, there have been Jukes-Cantor model and Kimural model.

Determine a tree building method. There are basically two types of phylogenetic methods, character based methods and distance based methods. Character-based methods based on discrete characters from molecular sequences from individual taxa. The theory is that characters at corresponding positions in a multiple sequence alignment are *homologous* (this word has different meaning in mathematics and is precisely defined for sequences with differentials.) among the sequences involved. On the other hand, distance-based methods is based on the *distance*, the degrees of differences between pairs of seqeuences. Such distance will be used to construct the distance matrix between individual pairs of taxa. The theory in this case is that all sequences involved are homologous and the weighted directed tree will satisfy the additive properties. There are two different algorithms in distance based method, the cluster-based and the optimality-based. The cluster-based method algorithms build a phylogenetic tree based on a distance matrix starting from the most similar sequence pairs. The algorithms of cluster-bsed include unweighted pair group method using arithmetic average (UPGMA) and neighbor joining (NJ). The optimality-based method algorithms compare numerous different tree topologies and select the one which is believed to best fit between computed distances in the trees and the desired evolutionary distances which often referred as *actual evolutionary distances.* Algorithms of optimality based include Fitch-Margoliash and minimum evolution.

Assess tree reliability. The assessment of constructed phylogenetic trees will be on the reliability of the inferred phylogeny. More precisely, is any particular tree more reliable than the others? Or, is it more biologically or statistically significant than the other? Bootstrapping and Jacknifing are two analytical strategies which repeatedly resample data from the original database to test for errors of a phylogenetic trees. Kishino-Hasegawa and Shimodaira-Hasegawa tests are often used to compare the significance of two phylogenetic trees.

In summary, finding correct and acurate phylogenetic trees is generally an extremely difficult task. One argument is to calculate how many different such trees exist, and these numbers are indeed astronomically large when the number of taxa increases. While the argument makes some mathematical sense, it makes absolutely no biological sense because many of the mathematically existing trees can be eliminated by biological common sense. It is more due to the facts that often times one has to derive a consensus tree from multiple individual trees based majority rule, or any other rules which would make biological sense in practice, and both *consensus* and these *rules* are often very fuzzy and varies from people to people. Moreover, at each step of the tree building, there are multiple choices, criteria to make those choices are so vague that different choices often lead to some significantly different outcomes.

A more detailed version of the introduction to the area can be found in references such as J. Xiong [7] and M. Nei *et al* [5]. Our focuses in this paper shall be the step 4 in the process, to determine and access tree building methods, in particular, the distance based methods. The paper starts with some preliminary knowledge and definitions in the area, including finite directed graphs, directed trees and matrices. It discusses the verification of the metric property of distance matrices, including detections of errors if a distance matrix fails to satisfy the metric property, and then provides an algorithm in modifying the distance matrix to satisfy the metric property. The second part of the paper is a brief survey based on the excerpts from the references, on various frequently used distance based phylogenetic tree construction methods, both cluster-based and optimality base methods, including UPGMA, Neighbor Joining, Fitch-Margoliash, and Minimum Evolution methods. Also, it discusses the assessment of the phylogenetic trees and some analysis of the algorithms.

2. Preliminaries

For the completeness of our discussion, we begin with some basic definitions.

Definition 2.1. *A finite* **graph** *G is a pair (V, E), where V is a finite set, and E is a subset of all two member subsets (non ordered pairs) of V. The members in V are referred as* **vertices** *or* **nodes**, *and the members in E are referred as* **edges** *or* **branches**. *A finite* **directed** *graph is a finite graph with edge set E be a subset of the cartesian product (ordered paris) $V \times V$. For any edge $v = (C, B) \in E$ in a directed graph, where $C, B \in V$, C is called an* **ancestor** *of B and B is called an* **offspring** *of C.*

It is common practice in biology that if there is a branch from one node to another node, the offspring is drawn higher (lower) than the ancestor node, instead of specifically defining the direction of the branches.

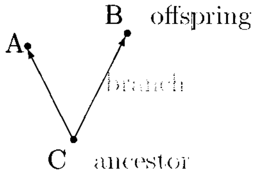

FIGURE 1. Directed graph

Definition 2.2. *A* **weighted** *graph is a graph with each edge in the graph has an assigned value.*

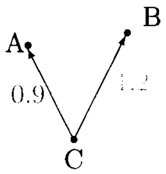

FIGURE 2. Weighted graph

There are several different ways to define a tree. A **circuit** in a finite graph is a sequence of vertices (nodes) $A_1, A_2, \ldots, A_n$, such that, for any i, $1 \leq i \leq n$, either (A_i, A_{i+1}) or (A_{i+1}, A_i) is an edge (branch). Same is true for (A_n, A_1) or (A_1, A_n). A **tree** is a finite graph without a circuit. A **phylogenetic tree** refers to a directed tree. A **rooted** tree is a directed tree such that there exists a node which is an ancestor of all other nodes, and this particular node is called **rooted node**. A **cladogram** is a directed (phylogenetic) tree. A **phylogram** is a weighted directed (phylogenetic) tree. If a node is not an ancestor of any other node, the node is called a **taxon** (*taxa* for plural).

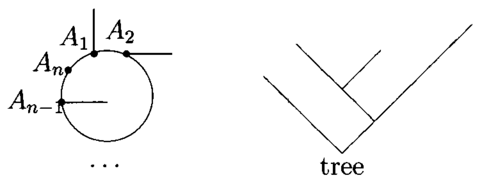

FIGURE 3. Circuit in graph and tree

A node has two offsprings is called **dichotomy**. A typical phylogenetic tree is a tree with only dichotomy, which is called *bifurcation* in biology. However, if a node with more than two offsprings is called **polytomy**, and referred as *multifurcation* in biology. Multifucation normally indicates there are some problems in fully resolving the phylogenetic tree or a result of an evolutionary process known as *radiation*. For the convenience of observation, the phylogenetic trees in bioinformatics community

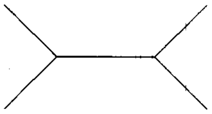

FIGURE 4. Unrooted tree

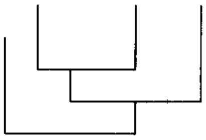

FIGURE 5. Phylogenetic tree in square form

are often drawn in square form. For example, the tree shown in figure 3 can be drawn as in figure 5.

When it comes to comparing tree, we define two phylogenetic trees $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are said to be **isomorphic** (same) if there exists a bijective map ϕ from V_1 to V_2, preserving the corresponding edges. Moreover, ϕ preserves the direction of all corresponding edges in the case of directed graphs, and weights in the case of weighted graphs. However, in real life, this kind of identity does not occur very often. Instead, we have to compare and evaluate two different trees. The **topological distance** between two different trees is defined as twice the number of interior edges (neither end is a taxon) at which sequence partition is different between the two trees. Clearly, if two phylogenetic trees are isomorphic, their topological distance is 0.

The core part in the distance-based methods to build phylogenetic trees is the introduction of distance between sequences. Let V be the underline set (of sequences, for instance). A **distance** is a function d from $V \times V$ to the set of non-negative real numbers $\mathbb{R}^+ \cup \{0\}$. Normally, the condition $d(A, A) = 0$ for any $d \in V$ is required. Moreover, a distance is said to be **symmetric** if $d(A, B) = d(B, A)$ for any two member $A, B \in V$. A distance is said to be **metric** if it satisfies the *triangular* inequality

$$d(A, C) \leq d(A, B) + d(B, C),$$

for any three members A, B and C in V. In fact, in mathematics, any measure is said to be *metric* if it is reflexive (for any A and B, $d(A, B) = 0$ if and only if $A = B$), symmetric, and satisfying the triangular inequality. In particular, a distance is said to be **additive** if the equality holds. That is,

$$d(A, C) = d(A, B) + d(B, C),$$

for any three members A, B and C in V.

Further, a distance is said to be **ultrasymmetric** if for any A, B and C in V, at least two of the three distances $d(A, B)$, $d(B, C$ and $d(C, A)$ are equal.

There are apparently many different ways to define distance. The most natural way would be the Hamming distance, introduced by Richard W. Hamming, in coding theory. The *Hamming distance* between two squences is the number of positions in which they differ. It is well known in mathematics that Hamming distance is metric.

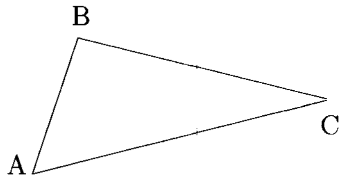

FIGURE 6. Triangular property

Moreover, since one can define algebraic structures (addition and subtraction, etc.) on the collection of all sequences, Hamming distance also satisfies the translation property. Namely, for any sequence C,

$$d(A,B) = d(A+C, B+C).$$

One commonly used distance in bioinformatics is to define the distance between any two nodes in an evolutionary phylogenetic tree based on observation as the sum of all weights along the path from one node to the other. Since there exists no circuit in a tree, this distance is well-defined and it is additive. In general, for any pair A, B of sequences, other than giving the evolutionary distance between two species based on observation, we can define the distance d_{AB} between the two sequences as the fraction f of sites u where residues x_u^A and x_u^B differ. However, this definition is no longer good for two unrelated sequences. Random substitutions will cause f to approach the fraction of differences, and it is expected the distance to go infinity as f goes to this value. Defining distance using Markov models of residue substitution, such as the Juke-Cantor model for DNA, will satisfy this desired property. The Jukes-Cantor distance is defined as

$$d_{AB} = -\frac{3}{4}\log\left(1 - \frac{4f}{3}\right).$$

d_{AB} goes to ∞ as f approaches to the equilibrium values of f (75% of residues different).

An important tool in distance-based methods in building phylogenetic tree is the use of distance matrix. An $m \times n$ **matrix** is a collection of nm real numbers, arranged in m rows and n columns.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ & & \cdots & & \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{bmatrix}$$

Each number a_{ij} in the matrix is called an *entry* of the matrix. A matrix is said to be *symmetric* if for any row and column i, j, $a_{ij} = a_{ji}$. The following examples show the symmetry and asymmetry of matrices.

$$\begin{bmatrix} 2 & 1 & 3 & -1 \\ 1 & 0 & 2 & 4 \\ 3 & 2 & 5 & 1 \\ -1 & 4 & 1 & -2 \end{bmatrix} \qquad \begin{bmatrix} 2 & 1 & 2 & -1 \\ 1 & 0 & 2 & 4 \\ 3 & 2 & 5 & 1 \\ -1 & 4 & 1 & -2 \end{bmatrix}$$

In particular, if a matrix has the same number of rows and columns, $n \times n$, it is often called a **square** matrix. Given a collection of n sequences, with distance d defined between any pair of sequences, the following matrix

$$\begin{bmatrix} d_{11} & d_{12} & d_{13} & \cdots & d_{1n} \\ d_{21} & d_{22} & d_{23} & \cdots & d_{2n} \\ d_{31} & d_{32} & d_{33} & \cdots & d_{3n} \\ & & \cdots & & \\ d_{n1} & d_{n2} & d_{n3} & \cdots & d_{nn} \end{bmatrix}$$

is call the **distance matrix**, where d_{ij} is the distance between the i^{th} and j^{th} sequences.

3. Building of Distance Matrix

One of chanllenges in using distance matrices with distance methods to build phylogenetic tree is the building of the matrix. The very necessary condition to make all the thory work is that the distance must satisfy metric property. That is, all the entries in the distance matrix representing distances between sequences must satisfy the triangulr inequality,

$$d_{ik} \leq d_{ij} + d_{jk},$$

for all $1 \leq i, j, k \leq n$. Unfortunately, many times it is not the case in practice. There are always some entries fail the inequality, if not all of them.

Our first algorithm is to correct any possible errors to make the matrix symmetric. Assuming that, upon detection of asymmetries, there is no way algorithm would know which one is the *true* correct data, it simply uses the average of the two values to substitute. The pseudo-code for the algorithm is given in the following.

```
Algorithm: Symmetry Enforcing - Iteration
Initialization:
  retrieve data matrix in a double string;
  n is the number of nodes, i=1;
Iteration:
  j=2;
  if d[ij]!=d[ji] then substitute d[ij] and d[ji]
    by (d[ij]+d[ji])/2;
  j=j+1; if j=n then i=i+1;
Termination:
  i=n.
```

The complexity of the algorithm is linear since the matrix has $O(n^2)$ entries.

3.1. **Metric distance matrix.** Assume we now have a symmetric distance matrix, our next challenge is to enforce the metric property. Again, our assumption is that the algorithm would not know the *true* correct data, it simply uses the minimum value to make the distance metric. More specifically, for a node A_n, if the triangular inequality holds, then for any $j, k < i$, we must have

$$d_{ij} \geq d_{jk} - d_{ik}.$$

Thus,

$$d_{ij} \geq \max_{k<j}\{\, d_{jk} - d_{ik} \,\}$$

is a necessary condition for the triangular inequality to be true. The pseudo-code for the algorithm is given in the following.

Algorithm: Metricalization of distance matrix

```
Initialization:
  retrieve data matrix M[n] in a double string;
  n is the number of nodes; i=3
Iteration:
  j=2;
  k=1;
  if d[ij]<d[jk]-d[ik] then substitute d[ij]
    by d[jk]-d[ik]
  k=k+1; if k=j then j=j+1;
  if j=i then i=i+1;
Termination:
  i=n+1.
```

The complexity of the algorithm will also be polynomial since

$$\sum_{i}^{n} i \cdot i(i+1) \cong O(n^4).$$

Also, the correction process is heavily based on the earlier portion of the data than the later portion. Namely, if a distance matrix fails the triangular property only at the last node or so, the correction is very minimum. On the other hand, if it occurs at the first three (automatically true with only two nodes) nodes, then it might have impact on all the data follows.

4. Clustering method

4.1. UPGMA.

UPGMA stands for Unweighted Pair Group Method using Arithmetic average. Given a distance matrix, it starts with grouping two taxa with the smallest distance between them according to the distant matrix. A new node is added in the midpoint of the two, and the two original taxa are put on the tree. The distance from the new node to other nodes will be the arithmetic average. We then obtain a reduced distance matrix by replacing two taxa with one new node. Repeat this process until all taxa are placed on the tree. The last taxon added will be the root of the tree. More precisely, for any two clusters C_i and C_j, we define the distance between the clusters as

$$d_{ij} = \frac{1}{|C_i||C_j|} \sum_{p \in C_i, q \in C_j} d_{pq},$$

where $|C_i|$ and $|C_j|$ denote the number of sequences in cluster i and j, respectively. The pseudo code of UPGMA is given in [2] as follows.

Algorithm: UPGMA

```
Initialization:
  assign each sequence i to its own cluster C[i];
  define 1 leaf for each sequence, place @ height 0
Iteration:
  determine the two clusters i, j with d[ij] min;
  define a new cluster k by C[k]=C[i]∪C[j];
  define node k with daughter nodes i and k;
  place the new node at height d[ij]/2;
  add k to current clusters and remove i and j;
Termination:
  when only two cluster i, j remain;
  place root at d[ij]/2.
```

In fact, the outcome phylogenetic tree from UPGMA is not an arbitrary tree. Instead, it is a special type of trees. It satisfies the *molecular clock* property. That is, the total time traveling down a path to the leaves from any node is the same, regardless the choice of path. Thus, if the original phylogenetic tree from observation does not have this property, it will not reconcile well with the tree from UPGMA. One necessary condition for producing better phylogenetic tree is the *ultrametric* condition, which requires any triangle must be at least equilibrium or equilateral, with regard to the weights between them.

UPGMA is actually a very simple algorithm. The search for the smallest distance takes complexity $n \log n$, with totals to $n^2 \log n$ complexity. There have been some variation of UPGMA, taking minimum or maximum distance of constituent sequences, instead of taking average, but none of those actually improves the performance.

4.2. **Neighbor Joining.** The UPGMA assumes unweighted distances and molecular clock property which means all taxa have constant evolutionary rates, and these normally may not be met in biological sequences. It is even possible that an additive tree fails to satisfy the molecular clock property. Neighbor joining method can be used in this circumstance. It uses the similar ideas to build a tree, but corrects unequal evolutionary rates between sequences by using a conversion step. Find a pair of neighboring leaves i and j which have the same parent node k. Remove them from the list of leaf nodes and add k to the current list of nodes, and define its distance to leaf m by

$$d_{km} = \frac{1}{2}(d_{im} + d_{jm} - d_{ij}).$$

By additivity, the distance d_{km} are exactly those between the corresponding nodes in the original tree. Repeat this process we can reduce the number by one at each iteration until we have a pair of leaves.

The following procedure is proposed by Saitou and Nei in 1987 and improved by Studier and Keppler in 1988. Define

$$D_{ij} = d_{ij} - (r_i + r_j),$$

where

$$r_i = \frac{1}{|L|-2}\sum_{k\in L} d_{ik}, \tag{4.1}$$

and $|L|$ denotes the size of the set L of leaves. The following result is proved by Studier and Keppler in 1988.

Theorem 4.1. *A pair of leaves i and j for which D_{ij} is minimal will be neighboring leaves.*

Proof. Suppose the smallest D is D_{ij}, and suppose furthermore that i and j are not neighboring leaves. Thus, there are must be at least two nodes on the path connecting them. Call these nodes k and l, and let L_k be the set of leaves which derive from the third branch from k, that is, not the edge towards i or j,and let L_l be the equivalent set for l. Let m and n be a pair of neighboring leaves in L_k with joining node p (if no such pair exists, an alternative argument will be available.) Let d_{uv} denote the summed edge lengths of the path connecting any two nodes u and v. By additivity, this is the correct distance d_{uv} when they are both leaves. For any $y \in L_k$, it is clear that

$$d_{iy} + d_{jy} = d_{ij} + 2d_{ky}.$$

Similarly,

$$d_{my} + d_{ny} = d_{mn} + 2d_{py}.$$

Thus,

$$d_{iy} + d_{jy} - d_{my} - d_{ny} = d_{ij} + 2d_{ky} - 2d_{py} - d_{nm}. \tag{4.2}$$

Similarly, for $z \in L_l$, we find

$$d_{iz} + d_{jz} - d_{mz} - d_{nz} = d_{ij} - 2d_{kp} - 2d_{lk} - d_{nm}. \tag{4.3}$$

From the definition of D_{ij},

$$D_{ij} - D_{mn} = d_{ij} - d_{mn} - \frac{1}{N-2}\left(\sum_{leavesu} d_{iu} + d_{ju} - d_{mn} - d_{nu}\right).$$

One can easily check from (4.2) and (4.3) that the coefficients of d_{ij} and d_{mn} summed over all leaves u in the tree, are both $N-2$. Thus, the term $d_{ij} - d_{mn}$ cancels each other, and we can write

$$D_{ij} - D_{mn} =$$

$$\frac{1}{N-2}\left(\sum_{y\in L_k}(2d_{py} - 2d_{ky}) + \sum_{z\in L_l}(2d_{pk} + d_{lk})\right) + C,$$

where C is the sum of all the extra positive terms coming from other branches on the path between i and j besides k and l. Let $|L_l|$ and $|L_k|$ denote the numbers of nodes in L_k and L_l, respectively, and use the fact that $d_{py} - d_{ky} > -d_{pk}$, we have

$$D_{ij} - D_{mn} > 2d_{pk}\frac{|L_l| - |L_k|}{N-2}.$$

Since D_{ij} is minimum, we must have $D_{mn} > D_{ij}$. Therefore, $|L_l| < |L_k|$. But the two nodes l and k are completely at the equal position, the very same argument can be applied with the two nodes l and k reversed. Thus, we must also have $|L_l| > |L_k|$. The contradiction concludes the proof.

The complete algorithm for neighboring joining works by constructing a tree, and keeping a list L of active nodes in this tree. If there were a pre-existing additive tree, L would be the current remaining set of leaf nodes as neighboring pairs were stripped away, and T would be the tree build up from these stripped-off nodes. The pseudo code is given as following.

Algorithm: Neighbor Joining

```
Initialization:
  define T the set of leaf nodes,
    one for each sequence
  L=T
Iteration:
  pick pair i, j in L for which D[ij] is minimal;     D[ij] is defined
in (4.1);
  define new node k;
  set d[km]=(d[im]+d[jm]-d[ij])/2 for all m in L;
  add k to T;
  set d[ik]=(d[ij]+r[i]-r[j])/2;
  set d[jk]=d[ij]-d[jk]-d[ij];
  join k to i, join k to j;
  remove i and j from L and add k;
Termination:
  when L consists of only 2 leaves i and j;
  add the remaining edge between i and j;
  set the length d[ij]
```

Unlike UPGMA, neighbor joining method produce unrooted trees. The outcomes has to be determined based on biological knowledge. We can use neighbor joining even the distance is not additive. Same as ultrametric property a test for molecular property, the following four point condition is a test for additivity. For any four leaves i, j, k, and l, two of the distances $d_{ij} + d_{kl}$, $d_{ik} + djl$, and $d_{il} + d_{jk}$ must be equal and larger than the thrid. This four-point condition is a necessary condition for additivity because two of the sums include the length of the bridge connecting pairs of leaves.

One of the disadvantages of the neighbor joining method is that it generates only one tree and does not test other possible tree topologies. This can cause problems because in many cases in the initial set up of neighbor joining, there might be multiple equally close pair of neighbors to join, leading to multiple trees. Since there is no way the algorithm would know which one is the most optimal tree, choosing an wrong option may produce a suboptimal tree. A generalized neighbor joining method has been developed in which multiple neighbor join trees with different initial taxon groupings are generated. A best tree can be selected based the actual evolutionary distances.

5. Optimality-based Methods

The clustering based methods produce a single tree as outcome. However, there is no criterion in judging how this tree is compared to other alternative trees. In contrast, optimality based methods have a well-defined algorithm to compare all possible tree topologies and select a tree that best fits the actual evolutionary distance matrix. Based on the differences in optimality criteria, there are two types of algorithm, Fitch-Margoliash and minimum evolution. The exhaustive search for an optimal tree necessitates a slow computation, which is a clear drawback especially when the dataset is large.

5.1. **Fitch-Margoliash.** The Fitch-Margoliash method selects a best tree among all possible trees based on minimal deviation between the distances calculated in the overall branches in the tree and the distances in the original dataset. It starts by randomly clustering two taxa in a node and creating three equations to describe the distances, and the solving the three algebraic equations for unknown branch lengths. The clustering of the two taxa helps to create a newly reduced matrix. This process is iterated until a tree is completely resolved. The method searches for all tree topologies and selects the one that has the lowest squared deviation of actual distances and calculated tree branch lengths. The optimality criterion is expressed in the following formula

$$E = \sum_{i=1}^{T-1} \sum_{j=i+1}^{T} \frac{(d_{ij} - p_{ij})^2}{d_{ij}^2},$$

where E is the error of the estimated tree fitting the original data, T is the number of taxa, d_{ij} is the pairwise distance between ith and jth taxa in the original dataset, and p_{ij} is the corresponding tree branch length.

5.2. **Minimum Evolution.** Minimum evolution constructs a tree with a similar procedure, but uses a different optimality criterion that finds a tree among all possible trees with a minimum overall branch length. The optimality criterion relies on the formula

$$S = \sum b_i,$$

where b_i is the ith branch length. Searching for the minimum total branch length is an indirect approach to achieving the best fit of the branch lengths with original dataset. Analysis has shown that minimum evolution in fact slightly outperforms the least square-base Fitch-Margoliash method.

The theoretical foundation of the minimum evolution method is the mathematical proof given by A. Rzhetsky and M. Nei [6] in 1993 showing that when unbiased estimates of evolutionary distances are used, the expected value of S becomes smallest for the true topology irrespective of the number of sequences. This is a good statistical property, but a topology with the smallest S is not necessarily an "unbiased estimator" of the true topology.

6. Assessment of Trees

6.1. **Bootstrapping.** The tree building algorithms produce a phylogenetic tree, or trees, but with no measure of how much they should be trusted. J. Felsenstein [3] in 1985 suggested that using the bootstrap as a method of assessing the significance of some phylogenetic feature, such as the segregation of a particular set of species on their own branch, known as *clade.*

The bootstrap works as follows: given a database consisting of an alignment of sequences, an artificial dataset of the same size is generated by picking columns from the alignment at random with replacement. (A given column in the original dataset can therefore appear several times in the artificial dataset.) The tree building algorithm is then applied to this new dataset, and the whole selection and tree building procedure is repeated some number of times, typically of the order of 1,000 times. The frequency with which a chosen phylogenetic feature appears is taken to be a measure of the confidence we can have in this feature.

For certain probabilistic models, the bootstrap frequency of a phylogenetic feature F can be shown to approximate the posterior distribution $P(F|\text{data})$. When the bootstrap is applied to a non-probabilistically formulated model, such as parsimony, it can be interpreted in terms of statistical hypothesis testing, though a rather more elaborate procedure than that given above may be needed to make the bootstrap conform to standard notions of confidence intervals.

6.2. **Jackknifing.** In addition to bootstrap, another often used resampling technique is jackknifing. In Jackknifing, one half of the sites in a dataset are randomly deleted, creating datasets half as long as the original. Each new dataset is subjected to phylogenetic tree construction using the same method as the original. The advantage of jackknifing is that sites are not duplicated relative to the original dataset and that computing time is much shortened because of shorter sequences. One criticism of this approach is that the size of dataset has been changed into one half and that the datasets are no longer considered replicates. Thus, the results may not be comparable with that from bootstrapping.

6.3. **Kishino-Hasegawa and Shimodara-Hasegawa Tests.** In phylogenetic analysis, it is also important to test whether two competing tree topologies can be distinguished and whether one tree is significantly better than the other. The task is different from bootstrapping in that it tests the statistical significance of the entire phylogeney, not just portions of it. For that purpose, several statistical tests have been developed specifically for each of the three types of tree reconstruction methods, distance, parsimony, and likelihood, including Kishino-Hasegawa and Shimodara-Hasegawa Test.

7. Analysis of the Algorithms

The most frequently used distance methods are cluster based. The major advantages is that they are computationally fast and are therefore capable of handling databases that are deemed to be too large for any other phylogenetic methods. The methods, however, are not guaranteed to find the best tree, whatever the *best* means. Exhaustive tree searching algorithms such as Fitch-Margoliash and Minimum Evolution have better overall accuracies. On the other hand, they can be computationally too expensive to use when the number of taxa is too large, even sometimes, only as large as 12, because overall number of topologically different trees will be exponentially growing and soon become too large to handle, even though nowadays the computing power has been increasing dramatically. A compromise between the two types of algorithms is a hybrid approach such as the generalized neighbor joining, with a performance similar to that of minimum evolution, but algorithm runs much faster.

The overall advantages of all distance based methods is the ability to make use of a large number of substitution models to correct distances. The drawback is that the actual sequence information is lost when all the sequence variation is reduced to a single value. Hence, ancestral sequences at internal nodes can not be inferred.

8. Acknowledgement

The author would like to sincerely thank Douglas Brutlag and his assistant Peggy Yao for offering such a well-organized course and all the helps and encouragements during the winter quarter, 2007. Also, many thanks to his colleagues, Joeseph McCray, Errol Archibold, and Chung Ng, without whom the author would not have understood a single word in this area.

References

[1] Doublas Brutlag. *Computational Molecular Biology*, Class lecture notes, Stanford University, Winter, 2007.

[2] R. Durbin *et al.* Biological Sequence Analysis *Probabilistic Modes of Proteins and Nucleic Acids.* Cambridge University Press, 1998, 11th ed, 2006.

[3] J. Felsenstein. *Confidence limits on phylogenies: an approach using the bootstrap.*, Evolution vol 39, 783-791.

[4] Joseph A. Gallian. *Contemporary Abstract Algebra*, Houghton Mifflin, 5th Ed, 2002.

[5] M. Nei and S. Kumar. *Molecular Evolution and Phylogenetics.* New York, Oxford University Press, 2000.

[6] A. Rzhetsky and M. Nei *Theoretical foundation of the minimum-evolution method of phylogenetic inference.* Mol. Biol. Evol. 10, 1073-1095.

[7] Jin Xiong. *Essential Bioinformatics*, Cambridge University Press, 2006.

Department of Mathematics, Morehouse College, Atlanta, GA 30314
E-mail address: `cpeng@@morehouse.edu, cpeng1@@stanford.edu`