Table of Contents

Introduction to Phylogenetics

Alignment of sequences should take account of their evolutionary relationship. For example, an alignment that implies many substitutions between closely related sequences is less plausible than one that makes most of its changes over large evolutionary distances.

The phylogenetic tree of a group of sequences does not necessarily reflect the phylogenetic tree of their host species, because gene duplication is another mechanism, in addition to speciation, by which two sequences can be separated and diverge from a common ancestor. If we are interested in inferring the phylogenetic tree of the species carrying genes, we must use orthologous genes (created by speciation events).

A true biological phylogeny has a 'root', or ultimate ancestor of all sequences. Some algorithms provide information, or at least a conjecture, about the location of the root. Others, like parsimony and the probabilistic models are completely uninformative about its position, and other criteria have to be used for rooting the tree.

Phylogeny

The evolutionary relationships among organisms; the patterns of lineage branching produced by the true evolutionary history of the organisms being considered.

Phylogenetics

Field of biology that deals with the relationships between organisms. It includes the discovery of these relationships, and the study of the causes behind their pattern.

Usefulness

- Infer function by similarity
- Choose template for homology modeling
- Discover and analyze families of genes
- Compare whole genomes

Taxon

A member of the groups of organisms being analyzed. This may be a single species or a group of species. It is the "label" at the leaf of the tree.

Phylogenetic Trees

Phylogenetic trees are most commonly binary trees. For n leaves, a rooted binary tree contains 2n-1 nodes and 2n-2 edges. An unrooted binary tree contains 2n-2 nodes and 2n-3 edges.

Homology

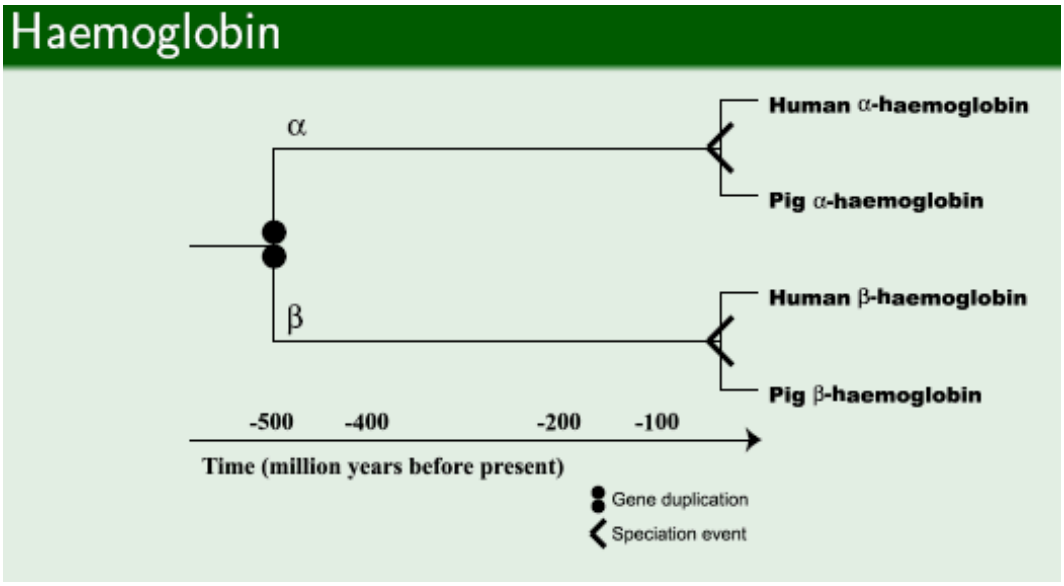
Similarity due to a common ancestor. It is in fact the hypothesis we make when we align sequences.

Homology is not similarity

Similarity is a measurable scale. Homology is a hypothesis that can be either true of false.

Orthologs vs. Paralogs

Two genes are orthologous if they diverged after a speciation event. Two genes are paralogous if they diverged after a gene duplication event.



Haemoglobin α and β are paralogs whether we compare within or across species. Human α-Haemoglobin and pig α -Haemoglobin are orthologs.

Comparing human α-Haemoglobin and pig β –Haemoglobin for the purpose of inferring function would give aberrant results.

Molecular Clock

At the molecular level, mutations occur with a certain probability. However, a date cannot be read directly from molecular data. In some organisms this rate is higher than others due to geographical and temporal variations. Mutations are not conserved at a constant rate. All purely molecular dating methods give aberrant results.

Homoplasy

The occurrence of similar states of a character not due to common lineage. This may be due to environmental constraints or simply a random occurrence.

Convergence: bats and birds have wings but don't share common ancestry.
Reversion: whales resemble fish but whale's ancestors lived on land.

Phylogenetic Algorithms

There are three classes of algorithms:

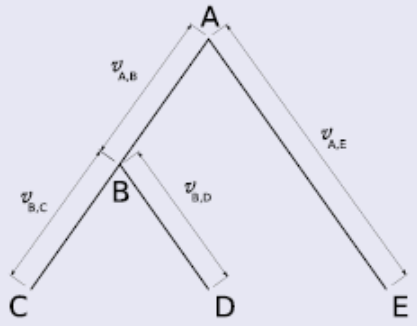
Numerical Taxonomic Phenetics

A distance score can be calculated between two taxons using PAM or BLOSUM alignment scoring. Clustering is another useful technique.

Cladistic Methods

Make inferences about characters at internal nodes. All cladistic methods attempt to find the minimal sum of all edges and vertices (minimize the length of the tree).

$$\min\left(\sum_{i \in edges} v_i\right)$$



The vast majority of cladistic methods are optimization algorithms. These algorithms search for an optimum in a search-space. The search space is the set of possible trees. This includes all topologies and all ancestral states for each topology.

A search methods could be brute-force, branch and bound or heuristic.

Brute Force cladistic search methods

The search space can be represented in the form of a tree. A selection is made at each node. In brute force, a complete search of all phylogenetic trees is made by walking the decision-tree and calculating the score at each leaf of the decision tree.

Branch and Bound cladistic search methods

Branch and bound algorithms also use search trees. The score of the partially constructed tree is calculated at each internal node. If the score is worse than the best score obtained so far, we do not continue with that branch.

Heuristic algorithm based cladistic search methods

Both brute force and branch and bound always find the best solution but they cannot do much in real time. Heuristic solutions are much faster but do not guarantee the optimal solution. Local optima vs. global optima.

Advantages of cladistic methods

- Take variable rates of evolution and homoplasy into account.
- Gives a tree with putative ancestral states.

Disadvantages of cladistic methods

- Slow

- Often only local optima is found
- Care must be taken when interpreting evolutionary distances
- Many equally optimal solutions may be generated

Probabilistic Methods

Probabilistic methods start with a model of evolution. This model is described in the form of mutation probabilities. The most probable tree given the data and the model can then be calculated. The probabilities of multiple mutations in a branch are also taken into account. The most commonly used probabilistic algorithms are maximum likelihood and bayesian methods.

Advantages

- Based on a model of evolution
- Take variable rates of evolution, homoplasy and even multiple mutations in a branch into account
- Statistical confidence for the result is inherent in the method

Disadvantages

- Slow
- Often only local optimum is found.

Miscellaneous Topics

Tree Rearrangement Algorithms

During a heuristic search, the local neighborhood of a solution must be searched in order to find local optimum. Since the search space cannot be defined in terms of Cartesian coordinates, a method of finding similar trees must be found. There are 3 methods to do so:

NNI - Nearest Neighbor Interchange

Take one branch and swap it with another.

SPR – Subtree Pruning and Regrafting

Chop off a piece of the tree and attach it somewhere else in the tree.

TBR - Tree Bisection and Reconnection

Bisect a tree and reconnect is differently.

Statistical Confidence

We do not always know whether we should trust the results of an analysis. Statistics help us evaluate the robustness of our results. If a small change in our data causes a large change in the results of our analysis, we should be careful.

The bootstrap method allows us to determine a statistical confidence level for any type of analysis as long as all the initial data-points follow the same distribution. We do not need to know the nature of the distribution.

Given a sequence where we do not know whether each amino acid is as important as the next, a change in the active site of an enzyme for instance represents a much larger evolutionary step than a change in a loop. The bootstrap method simply allocates weights in a random fashion.

Bootstrap works as follows;

1. Given a dataset 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.
2. The tree building algorithm is then applied to this new dataset, and the whole selection and tree building procedure is repeated, typically a thousand times.
3. The frequency with which a chosen phylogenetic feature appears is taken to be a measure of confidence we have in this feature.

When bootstrap is applied to a non-probabilistically formulated model, such as parsimony, it can be interpreted in terms of statistical hypothesis needed to make the bootstrap conform to standard notions of confidence intervals.

Making a tree from pairwise distances

One of the easiest to understand algorithms for tree drawing is the pairwise distance method. This method produces a rooted tree. The algorithm is initialized by defining a matrix of distances between each pair of sequences in the input set. Sequences are then clustered according to distance, in effect building the tree from the branches down to the root.

Distances can be defined by more than one measure, but one of the more common and simple measures of dissimilarity between DNA sequences is Jukes-Cantor distance, which is logarithmically related to the fraction of sites at which 2 sequences in an alignment differ. The Jukes-Cantor distance is scaled such that it approaches infinity as the fraction of unmatched residue pairs approaches 75%.

$$d_{ij} = -\tfrac{3}{4}\log(1 - 4f/3)$$

Clustering Methods: UPGMA

UPGMA stands for unweighted pair group method using arithmetic averages. It is a simple and intuitively appealing method. It works by clustering sequences, at each stage amalgamating two clusters and at the same time creating a new node on a tree.

To begin with, each sequence is assigned to its own cluster, and a branch or leaf of a tree is started for that sequence at height 0 in the tree. Then, the 2 clusters that are the closest

together in terms of whatever distance measure has been chosen are merged into a single cluster. A branch point or node is defined that connects the two branches. The node is placed at a height in the tree that reflects the distance between the two leaves that have been joined. This process is repeated iteratively, until there are only two clusters left. When they are joined, the root of the tree is defined. The branch lengths in a tree constructed using this process theoretically reflects evolutionary time. Refer to figure 7.4

Molecular clocks and the ultrametric property of distances

UPGMA produces a rooted tree of a special kind. The edge lengths in the resulting tree can be viewed as times measured by a molecular clock with a constant rate. The divergence of sequences is assumed to occur at the same constant rate at all points in the tree, which is equivalent to saying that the sum of times down a path to the leaves from any node is the same, whatever the choice of path. If our distance data are derived by adding up edge lengths in a tree T starting from the level of the leaves, each time a node is crossed, the distances of all leaves in the left branch from that node to the leaves will be the current minimum distance, and a node will therefore be added precisely where the node is encountered in the tree T.

If the original tree is does not act as above, it may have been constructed incorrectly by UPGMA. This happens if the closest leaves are not neighboring leaves, in other words, they don't belong to the same ancestor.

The ultrametic condition tests the correctness of the reconstruction. The distances dij are said to be ultrametric if, for any triplet of the sequences, xi, xj, xk, the distances dij, djk, dik are either equal, or two are equal and the remaining one is smaller. This condition holds for distances derived from a tree with a molecular clock.

Additivity and neighbor-joining

Given a tree, its edge lengths are said to be additive if the distance between any pair of leaves is the sum of lengths of the edges on the path connecting them. This property is built into UPGMA. However, it is possible for the molecular clock property to fail but for additivity to hold. In such a case, there are algorithms which can be used to reconstruct the tree.

Given a tree with additive lengths d, we reconstruct it from the pairwise distances of it leaves dij as follows:

1. find a pair of neighboring leaves that have the same parent node k
2. Remove them from the list of leaf nodes
3. Add k to the current list of nodes, defining its distance to leaf m by the below mentioned formula
4. Repeat until you reach the pair of leaves

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

Simply said, you are adding nodes and additive distances until you can insert the leaves.

Additivity is a property that depends on the distance measure used. A tree may be additive with respect to one distance measure and not with respect to another. Additivity means that the sums of 2 lengths must be larger than a third and equal in size.

Neighbor-joining involves stripping all nodes with the exception of pre-existing additive trees. The tree is then reassembled.

Rooting Trees

Unlike UPCMA, neighbor-joining produces unrooted trees. Finding the root is a secondary task, which can be accomplished by adding an outgroup, or species that is known to be distantly related. In the absence of an outgroup, ad hoc strategies such as picking the midpoint are used.

Parsimony

Parsimony is the most widely used of all tree building algorithms. It works by finding the tree which can explain the observed sequences with a minimal number of substitutions. Instead of building a tree, it assigns a cost to a given tree, and it is necessary to search through all topologies, or to pursue a more efficient strategy that achieves this effect in order to identify the best tree. We can therefore distinguish two components to the algorithm:

1. Computation of a cost for a given tree T
2. a search through all trees, to find the overall minimum of this cost.

Parsimony treats each site independently, and then adds the substitutions for all sites. The basic step is therefore counting the minimal number of changes that need to be made at one site, given a topology and an assignment of residues to the leaves.

The weighted parsimony algorithm doesn't just count the number of substitutions, but adds cost S(a,b) for each substitution of a by b; the aim is now to minimize this cost. The algorithm starts at the leaves and works its way up to the root. This way of passing through a tree is called post-order traversal.

It is sometimes of interest to find the ancestral assignments of residues that give the minimal cost. For instance, one way of defining a length for an edge is to count the number of mismatches along the edge that occur in all possible minimal-cost ancestral assignments to the tree. This is achieved by keeping the lowest cost paths to daughter nodes.

In traditional parsimony, we just count the number of substitutions. All that is needed to obtain the cost of the tree is to keep a list of minimal cost residues at each node, together with the current cost C.

Parsimony algorithm can be used with both rooted and unrooted trees, however, it is easier to count rooted trees since the root defines a direction.

Selecting labeled branching patterns by branch and bound

With parsimony, the number of topologies swiftly becomes large as the number of leaves increase. Therefore, more efficient search strategy is needed than simple enumeration.

Some tree-searching methods proceed stochastically. For instance, we can swap randomly chosen branches on a tree and choose the altered tree if it better than the current one. This, however, does not guarantee to find the overall best tree. Another strategy is to build up the tree by adding edges one at a time. We then continue to add sequences with best scoring edges until the tree is complete. This method also fails to yield the overall best tree.

Parsimony exploits the fact that the number of substitutions in a tree can only be increased by adding an edge. The idea behind branch and bound is to begin systematically building trees with increasing number of leaves, but to abandon a particular avenue of tree building whenever the current incomplete tree has a cost exceeding the smallest cost obtained so far for a complete tree.

Simultaneous alignment and phylogeny

Let's consider the problem of simultaneously aligning sequences and finding a plausible phylogeny for them. There are 2 parsimony-type algorithms that tackle this problem, the first using a character-substitution model of gaps, the second using affine gap penalties. Both find an optimal alignment given a tree and both require a search over trees to find the overall optimum.

Sankoff & Cedergren's gap substitution algorithm

This method simultaneously produces an MSA and phylogenetic tree for nucleotide sequences. This method uses a maximum parsimony calculation in conjunction with a scoring function that penalizes gaps and mismatches, thereby favoring the tree that introduces a minimal number of such events. The inputed sequences of the interior nodes of the tree are scored and summed over all the nodes in each possible tree. The lowest-scoring tree sum provides both an optimal tree and an optimal MSA given the scoring function. Because the method is highly computationally intensive, an approximate method in which initial guesses for the interior alignments are refined one node at a time. Both full and approximate versions are in practice calculated by dynamic programming.

Hein's affine cost algorithm

Affine gap cost: a scoring system for gaps within alignments that charges a penalty for the existence of a gap and additional per-residue penalty proportional to the gap's length. Hein's algorithm uses an affine gap cost which is more realistic than the simple substitution treatment of gaps. It is also much faster than Sankoff & Cedergren's algorithm in the most realistic situations, fast enough in fact to allow a search over tree topologies for modest-sized sets of sequences. It is currently the only practical algorithm

able to align sequences and explore alternative phylogenies effectively. The price paid for these very considerable gains is that the algorithm makes a simplifying assumption in the choice of ancestral sequences which does not always lead to the overall most parsimonious choices.

Suppose we are given a tree. Recall that the algorithm for traditional parsimony ascends the tree, assigning a list of possible residues to each node. The residues are just those that minimize the number of substitutions along the edges to the two daughter nodes. In this case it is possible to find the minimal number of substitutions for the whole tree by minimizing at each node. This same procedure is used in Hein's algorithm. In the upward pass through the tree, only the minimum cost sequences at each node are considered. Unlike traditional parsimony, this procedure is not guaranteed to find the minimum cost for the whole tree.

The aim is to find sequences z

Review