The comparison of phylogenetic reconstruction using five different methods

Lívia Qian
KTH Royal Institute of Technology
School of Electrical Engineering and Computer Science
liviaq@kth.se

Abstract

The aim of this paper is to briefly describe and compare the performance of some widely accepted and one lesser-known method for phylogenetic reconstruction. Instead of trying to be a comprehensive guide summarizing the theoretical background of phylogenetics, it focuses on the main principles of creating simple phylogenetic trees and the conclusions I reached while testing these methods. While my primary goal was not to optimize them and making them scale to more complex data, approximating the time complexity of each algorithm was a crucial part of the experiments besides measuring the accuracy of the generated trees.

Introduction

Phylogenetics is the study of the evolutionary history of species and groups of species. There are multiple forms of expressing the relationship between different groups, one of them being the phylogenetic tree (or evolutionary tree), a diagram containing all the taxa in the form of leaves. Common traits can be observed using DNA sequences or morphology (the latter being generally more complex due to the many characteristics I certain species can have). Phylogenetic trees can be rooted and unrooted; the former is a version representing not only the similarity between each taxon but the ancestral relationships as well, while the latter form only illustrates the connection between the taxonomical units.

Methods

There are multiple steps included in creating a phylogenetic tree; DNA sequences need to be of the same length, therefore a proper sequence alignment is needed before most of the algorithms can be used. There are three main categories of tree-creating algorithms: distance-based, maximum parsimony and maximum likelihood methods. While all these techniques are fairly popular, distance methods are usually recommended when computational time is an important factor; in other cases, parsimony and likelihood are preferred as they are more rigorous and have the ability to explore different combinations.

Sequence alignment

DNA sequencing is the process of determining the order of nucleoids in the DNA; there are four bases — adenine, guanine, cytosine, and thymine — represented by characters A, G, C, T whose order needs to be determined in order to gain relevant information about the molecule at hand. Sequence alignment is a way of arranging multiple DNA sequences into a matrix so that similar regions are grouped together — that is, regions that show a high degree of similarity in multiple sequences are placed under-

neath each other. As this process makes the characters in the sequences shift, a special character (usually a "-") is used to fill in the gaps.

Neighbor joining

Neighbor joining is one of the distance-based methods created by Saitou and Sei in 1987 [1].

UPGMA

Originally attributed to Sokal and Michener [2], this method uses...

WPGMA

Maximum parsimony

Maximum parsimony is an umbrella term for all the methods that...

Maximum likelihood

Maximum likelihood, like in many other cases where probabilistic methods can be used, makes use of the basics of Bayesian statistics and looks at the probability of a certain sequence given a model (the model can be freely chosen). It may have a high algorithmic complexity as evaluating one sequence in itself may be computationally intensive, let alone multiple sequences. There are Bayesian methods that build upon ML; the major difference between these two groups of tree-building algorithms is that Bayesian methods take prior knowledge into account.

Self-growing tree algorithm or self-organizing tree algorithm

This one is based on a paper published in 1997 by Dopazo and Carazo [3] and is a combination of the Kohonen self-organizing map [4] and the growing cell structures algorithm of Fritzke [5]. This is an unsupervised learning network that starts out as a tree consisting of a small number of nodes (the paper mentioned two sister nodes) and then alternates between growing and adapting to the input sequences until it is fully grown and every taxonomical unit is assigned a proper place. The strictest exit condition guarantees that every input sequence is associated to a unique cell.

First, the input sequences need to be converted to one-hot encoding. Secondly, the tree's initial node(s) and the corresponding weight matrices that the encoded input sequences will ultimately be compared to need to be created; in order to distinguish inner nodes from leaves, the authors of the paper mention that it would be best to call them nodes and cells, respectively. In each step, nodes are considered "closed", meaning that they cannot be assigned sequences after they transition from being a cell to a node. After the first few cells are initialized with numbers ranging from 0 to 1, the alternating phases begin to take place. The first phase is called adaptation, which is basically what characterizes Kohonen's self-organizing maps: the input points are compared to all available cells and those that are closest to each of the inputs are updated, along with their neighborhood. After running this for a number of epochs, the inputs are mapped to appropriate positions in the output space, that is, the cells that they are the closest to according to a predefined metric, which is...

$$\frac{1}{2} \tag{1}$$

The concept of neighborhood is trickier than in the case of a 1D or 2D output space; ...

The second phase is the growing of the tree. As it is mentioned in the paper, ...

The tree stops growing when the resource value of the cell with the highest value is smaller than a predefined threshold — this threshold is zero if the goal is to map each sequence to a unique cell, or a sufficiently small number, in order to avoid inaccuracies in numerical calculations. The concept of resource value is related to the distance between cells and input points, and is defined as...

$$\frac{1}{2} \tag{2}$$

The criteria used for monitoring the convergence of the network... This helps in setting an exit condition for the adaptation process (e.g., the process can be ended when the relative increase of the error falls below a small threshold).

What is also interesting is the update rule...

Implementation

I implemented the algorithms mentioned in the previous section using Python in combination with NumPy and Pandas. For reference and sanity tests, I used Biopython's implementation of some of the algorithms mentioned above. Since some of the data structures needed were not feasible to implement within the scope of the project, I decided to use Biopython's corresponding classes to facilitate the work. These consist of `Bio.SeqIO`, `Bio.Align.MultipleSeqAlignment`, `Bio.Alphabet` and `Bio.Phylo.BaseTree`. These were needed to read in data, create sequence alignments, fill in the gaps in the sequence alignments and create the trees in a form that can be visualized easily, respectively.

Regarding phylogeny, I decided to use unrooted trees because they are generally more accurate and more easily comparable in a lot of cases. Biopython's `BaseTree` can easily be visualized with `Bio.Phylo.draw`, a function that takes into account features like branch length and custom labels.

The methods I implemented are neighbor joining, UPGMA, WPGMA, maximum parsimony and SOTA. In the case of maximum parsimony, ... The self-organizing tree algorithm relies on a couple of hyperparameters; these need to be tuned before extensive testing.

Data

Experiments

Hyperparameters of SOTA

Comparison with regard to running time

Comparison with regard to accuracy

References

[1] The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, July 1987.

[2] R. R. Sokal and C. D. Michener. A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38:1409–1438, 1958.

[3] Joaquín Dopazo and José María Carazo. Phylogenetic Reconstruction Using an Unsupervised Growing Neural Network That Adopts the Topology of a Phylogenetic Tree. *Journal of Molecular Evolution*, 44(2):226–233, February 1997.

[4] Teuvo Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990.

[5] Bernd Fritzke. Growing cell structures—a self-organizing network for unsupervised and supervised learning. *Neural Networks*, 7(9):1441–1460, January 1994.

[6] Alexandre De Bruyn, Darren P. Martin, and Pierre Lefeuvre. Phylogenetic reconstruction methods: An overview. In *Methods in Molecular Biology*, pages 257–277. Humana Press, December 2013.

[7] Arjun B. Prasad, Marc W. Allard, and Eric D. Green and. Confirming the phylogeny of

mammals by use of large comparative sequence data sets. *Molecular Biology and Evolution*, 25(9):1795–1808, May 2008.

[8] Martin Vingron, Jens Stoye, Hannes Luz, and Roland Wittler. Algorithms for phylogenetic reconstructions. February 2002-2009. Lecture notes from Bielefeld University.