Neighbor Joining with Subtree Weights

D. Levy[1], R. Yoshida[2], and L. Pachter[1]

[1]University of California at Berkeley, Berkeley, CA 94720, USA; [2]University of California at Davis, One Shields Ave. Davis, CA 95616, USA

The software package `MJOIN` will be available at `http://bio.math.berkeley.edu/mjoin/`

Introduction

- The Neighbor-Joining algorithm is a recursive procedure to reconstruct a phylogenetic tree using a transformation of pairwise distances between leaves for identifying cherries in the tree.
- Pachter and Speyer showed that we can recover an n-leaf tree from the weights of m-leaf subtrees if $n \geq 2m-1$ [PS04].
- We generalized the cherry picking criterion with estimates of the weights of m-leaf subtrees.
- We showed that a reconstructed tree from such weights is more accurate than one using pairwise distances.
- This leads to an improved neighbor-joining algorithm whose total running time is still polynomial in the number of taxa.

Neighbor Joining with Pairwise Distances

Theorem. (the cherry picking criterion) [SN87, SK88]
Suppose $D(ij)$ is a pairwise distance between taxa i and j. Then, $\{i,j\}$ is a cherry if $A_{ij} = D(ij) - (r_i + r_j)/(n-2)$, where $r_i := \sum_{k=1}^{n} D(ik)$, is minimal.

Idea. Initialize a star-like tree and find a cherry. Then we compute branch length from the interior node to each leaf. Repeat this process recursively until we find all cherries.

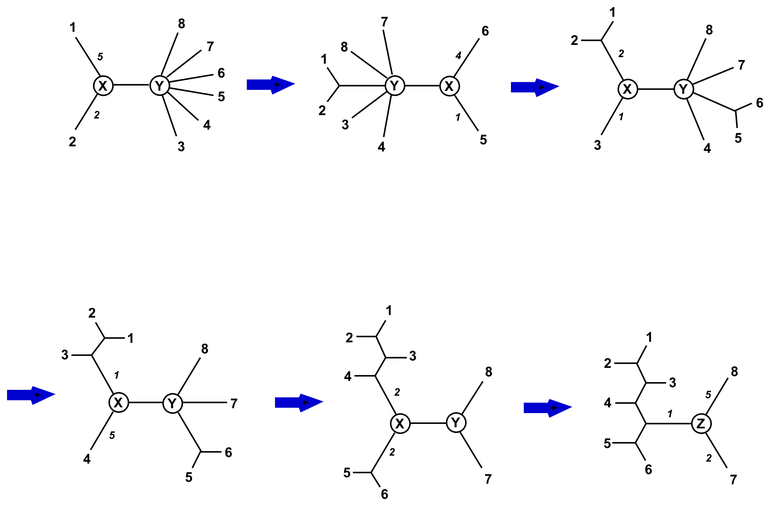


FIGURE 1: The traditional Neighbor Joining with pairwise distances.

Neighbor Joining with Subtree Weights

Notation. Let $[n]$ denote the set $\{1, 2, ..., n\}$ and $\binom{[n]}{m}$ denote the set of all m-element subsets of $[n]$.
Definition. A *m-dissimilarity map* is a function $D : \binom{[n]}{m} \to \mathbb{R}_{\geq 0}$. In terms of phylogeny, this corresponds to the weights of m-subtree weights of a tree T.
Theorem. Let D_m be be an m-dissimilarity map on n leaves, $D_m : \binom{[n]}{m} \to \mathbb{R}_{\geq 0}$ correspond to the weights of m-subtree weights of a tree T and we define

$$S(ij) := \sum_{X \in \binom{[n]\setminus\{i,j\}}{m-2}} D_m(ijX).$$

Then $S(ij)$ is a tree metric.
Furthermore, if T' is the additive tree corresponding to this tree metric then T' and T have the same tree topology and there is an invertible linear map between their edge weights.

Algorithm. (Neighbor Joining with Subtree Weights)

- **Input**: n many DNA sequences.
- **Output**: A phylogenetic tree T with n leaves.

1. Compute all m-subtree weights via the maximum likelihood.
2. Compute $S(ij)$ for each pair of leaves i and j.
3. Apply Neighbor Joining method with a tree metric $S(ij)$ and obtain additive tree T'.
4. Using a linear mapping, obtain a weight of each internal edge and each leaf edge of T.

Cherry Picking Theorem

Theorem. Let T be a tree with n leaves and no nodes of degree 2 and let m be an integer satisfying $2 \leq m \leq n-2$. Let $D : \binom{[n]}{m} \to \mathbb{R}_{\geq 0}$ be the m-dissimilarity map corresponding to the weights of the subtrees of size m in T. If $Q_D(ab)$ is a minimal element of the matrix

$$Q_D(ab) = \binom{n-2}{m-1} \sum_{X \in \binom{[n]\setminus\{i,j\}}{m-2}} D(ijX) - \sum_{X \in \binom{[n]\setminus\{i\}}{m-1}} D(iX) - \sum_{X \in \binom{[n]\setminus\{j\}}{m-1}} D(jX)$$

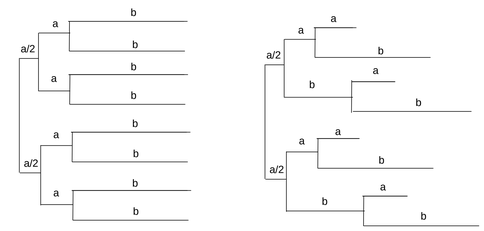
then $\{a, b\}$ is a cherry in the tree T.
Note. The theorem by Saitou-Nei and Studier-Keppler is a corollary from Cherry Picking Theorem.

Time Complexity

If $m \geq 3$, the time complexity of this algorithm is $O(n^m)$, where n is the number of leaves of T and if $m = 2$, then the time complexity of this algorithm is $O(n^3)$.
Note: The running time complexity of the algorithm is $O(n^3)$ for both $m = 2$ and $m = 3$.

Computational Results

We generate 500 replications with the Jukes-Cantor model via a software `evolver` from `PAML` package.



T1 T2

FIGURE 2: Modeled from Strimmer and von Haeseler.

The number represents a percentage which we got the same tree topology. l is the length of sequences.

l	a/b	m=2	m=3	m=4	fastDNAml
500	0.01/0.07	68.2	76.8	80.4	74.8
	0.02/0.19	54.2	61.2	73.6	55.6
	0.03/0.42	10.4	12.6	23.8	12.6
1000	0.01/0.07	94.2	96	97.4	96.6
	0.02/0.19	87.6	88.6	96.2	88
	0.03/0.42	33.4	35	52.4	33.6

The table above represents success rates for the model T_1. We compared our method with `fastDNAml` [HO94].

l	a/b	m=2	m=3	m=4	fastDNAml
500	0.01/0.07	84.4	86	85.6	88.4
	0.02/0.19	68.2	72	73.2	88.4
	0.03/0.42	18.2	29.2	36.2	87.4
1000	0.01/0.07	95.6	97.8	97.4	99.4
	0.02/0.19	88.4	89.6	93.4	99.8
	0.03/0.42	40	48.2	57.6	96.6

The table above represents success rates for the model T_2. We compared our method with `fastDNAml` [HO94].

References

[HO94] G. J. Olsen H. Matsuda R. Hagstrom and R. Overbeek. fastdnaml: A tool for construction of phylogenetic trees of dna sequences using maximum likelihood. *Comput. Appl. Biosci.*, 10:41–48, 1994.

[PS04] L. Pachter and D. Speyer. Reconstructing trees from subtree weights. *Applied Mathematics Letters*, 17:615 – 621, 2004.

[SK88] J. A. Studier and K. J. Keppler. A note on the neighbor-joining method of saito and nei. *Mol. Biol. Evol.*, 5:729 – 731, 1988.

[SN87] N. Saitou and M. Nei. The neighbor joining method: a new method for reconstructing phylogenetic trees. 1987.