Partitioning the Sample Space on Five Taxa for the Neighbor Joining Algorithm

Ruriko Yoshida
Dept. of Statistics University of Kentucky

Joint work with Kord Eickmeyer

www.ms.uky.edu/~ruriko

Future Work

The Minimum Evolution (ME) method: This is a **distance based method** and weighted Least Square method. It finds a closest additive metric from the given non-additive distance matrix with the smallest branch lengths (more biologically makes sense). Its time complexity is NP-hard.

Neighbor-joining (NJ) algorithm: This is the most popular distance based method. It estimates the ME tree (it is a greedy algorithm to find the ME tree). Its time complexity is polynomial time ($O(n^3)$).

Question: From this point of view, the NJ is "optimal" when the algorithm outputs the ME tree. Then, how "often" the NJ returns the ME tree (i.e., the output tree from the ME and the output tree from the NJ have the same tree topology)? This is joint work with K. Eickmeyer, P. Huggins, and L. Pachter.

The NJ phylogenetic tree for large data sets loses so much sequence information and we do not know how well it performs with pairwise distances that are not tree metrics, especially when all pairwise distances are estimated via the MLE.

Goal:

- Analyze the behavior of the NJ on five/six taxa.
- Show that the NJ tree topology is determined by polyhedral subdivisions of the spaces of dissimilarity maps $\mathbb{R}_+^{\binom{n}{2}}$.

Notation: We notate a, b, c, d as leaves and i, j as a pair of leaves. ($i = \{a, b\}$ etc...)

Neighbor Joining algorithm

Def. We call a pair of two distinct leaves $\{a, b\}$ a **cherry** if there is exactly one intermediate node on the unique path between a and b.

Let $D = (D(ab)) \in \mathbb{R}^{n \times n}$ be the distance matrix of T.

Thm. (Q-criterion) [Saitou-Nei, 1987 and Studier-Keppler, 1988]

Let $Q \in \mathbb{R}^{n \times n}$ such that $Q_{ab} = D_{ab} - (r_a + r_b)/(n-2)$, where $r_a := \sum_{k=1}^{n} D_{ak}$. $\{a^*, b^*\}$ is a cherry in T if $Q_{a^*b^*}$ is a minimum for all a and b.

Neighbor Joining Algorithm:

Input. A tree matric D. **Output.** An additive tree T.
Idea. Initialize a star-like tree. Then find a cherry $\{a, b\}$ and compute branch length from the interior node x to a and from x to b. Repeat this process recursively until we find all cherries.

The Q-criterion

The resulting matrix is again symmetric, and we can see it as a vector of dimension $m = \binom{n}{2}$ just like the input data. Moreover, the Q-criterion is obtained from the input data by a linear transformation:

$$\mathbf{q} = \mathbf{A}^{(\mathbf{n})}\mathbf{d},$$

where $\mathbf{d}$ is a vector representation of D, $\mathbf{q}$ is a vector representation of Q, and the entries of the matrix $A^{(n)}$ are given by

$$\mathbf{A}^{(\mathbf{n})}_{\mathbf{ij}} = \mathbf{A}^{(\mathbf{n})}_{\mathbf{ab,cd}} = \begin{cases} n-4 & \text{if } i = j, \\ -1 & \text{if } i \neq j \text{ and } \{a,b\} \cap \{c,d\} \neq \emptyset, \\ 0 & \text{else,} \end{cases}$$

where $a > b$ is the row/column-index equivalent to i and likewise for $c > d$ and j.

Example

For $n = 4$ we have

$$A^{(4)} = \begin{pmatrix} 0 & -1 & -1 & -1 & -1 & 0 \\ -1 & 0 & -1 & -1 & 0 & -1 \\ -1 & -1 & 0 & 0 & -1 & -1 \\ -1 & -1 & 0 & 0 & -1 & -1 \\ -1 & 0 & -1 & -1 & 0 & -1 \\ 0 & -1 & -1 & -1 & -1 & 0 \end{pmatrix}.$$

The Q-criterion:

find smallest $\mathbf{q_i}$ for $\mathbf{i = 1, \cdots, m}$ such that $\mathbf{q = A^{(n)} d}$.

Shifting Lemma

Note: There is an n-dimensional linear subspace of $\mathbb{R}^m$ which does not affect the outcome of NJ (Mihaescu et al, 2006). For a leaf a we define its *shift vector* $\mathbf{s}_a$ by

$$(\mathbf{s_a})_{\mathbf{b,c}} := \begin{cases} 1 & \text{if } a \in \{b, c\} \\ 0 & \text{else} \end{cases}$$

which represents a tree where the leaf a has distance 1 from all other leaves and all other distances are zero. The Q-criterion of any such vector is -2 for all pairs, so adding any linear combination of shift vectors to an input vector does not change the relative values of the Q-criteria.

The first step in cherry picking

After computing the Q-criterion $\mathbf{q}$, the NJ proceeds by finding the minimum entry of it, or, equivalently, the maximum entry of $-\mathbf{q}$.

Therefore, the set of all *parameter* vectors $\mathbf{d}$ for which the NJ will select cherry i in the first step is the normal cone at a vertex $-Ae_i$ of the polytope

$$\mathbf{P_n} := \text{conv}\{-\mathbf{Ae_1}, \ldots, -\mathbf{Ae_m}\}. \tag{1}$$

The shifting lemma implies that the affine dimension of the polytope P_n is at most $m - n$.

Reducing the number of taxa

Suppose out of our n taxa $\{1, \ldots, n\}$, the first cherry to be picked is the $\binom{n}{2}$th cherry $\{n-1, n\}$, which we view as the new node number $n-1$.

The reduced pairwise distance matrix is one row and one column shorter than the original one. Explicitly,

$$\mathbf{d'_i} = \begin{cases} d_i & \text{for } 1 \leq i \leq \binom{n-2}{2} \\ \frac{1}{2}(d_i + d_{i+(n-2)} - d_{m-1}) & \text{for } \binom{n-2}{2} + 1 \leq i \leq \binom{n-1}{2} \end{cases}$$

We see that the reduced distance matrix depends linearly on the original one:

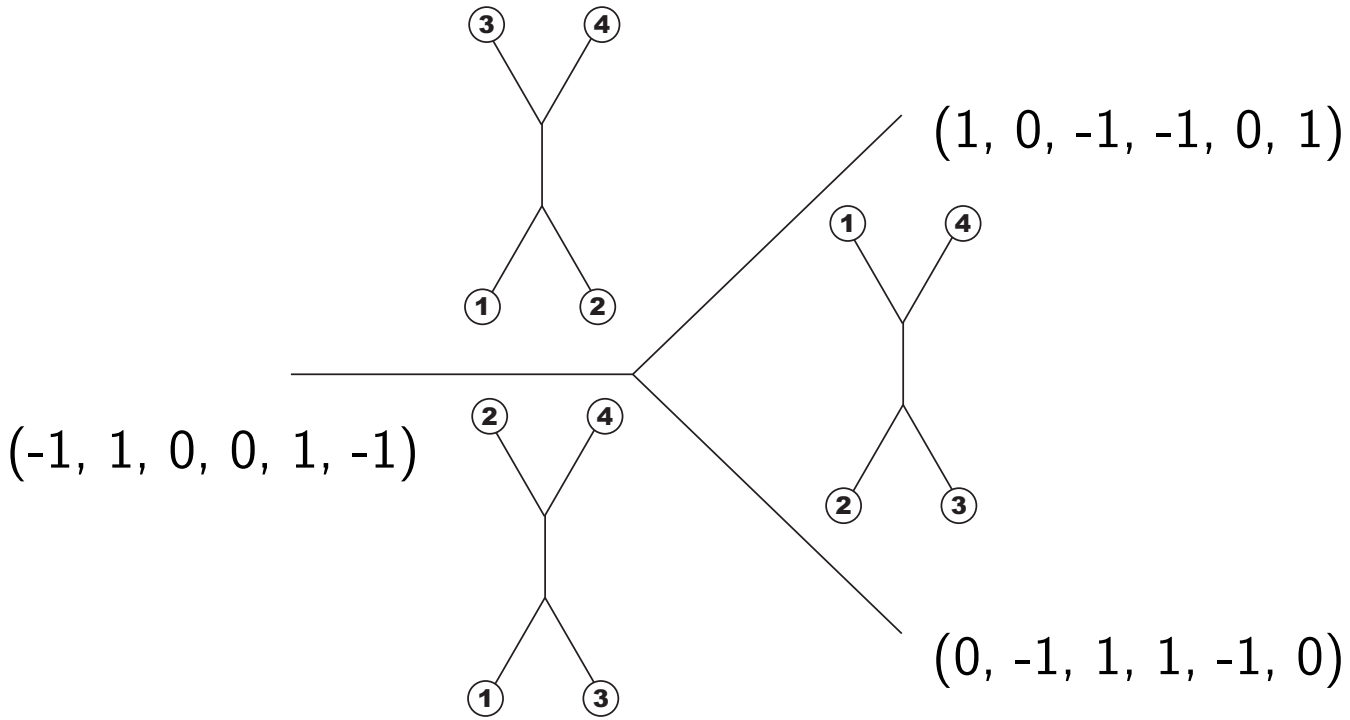
$$\mathbf{d'} = \mathbf{R}\mathbf{d},$$

with $R = (r_{ij}) \in \mathbb{R}^{(m-n+1)\times m}$, where

$$r_{ij} = \begin{cases} 1 & \text{for } 1 \leq i = j \leq \binom{n-2}{2} \\ 1/2 & \text{for } \binom{n-2}{2} + 1 \leq i \leq \binom{n-1}{2}, j = i \\ 1/2 & \text{for } \binom{n-2}{2} + 1 \leq i \leq \binom{n-1}{2}, j = i + n - 1 \\ -1/2 & \text{for } \binom{n-2}{2} + 1 \leq i \leq \binom{n-1}{2}, j = m \\ 0 & \text{else} \end{cases}$$

The process of picking cherries is repeated until there are only three taxa left, which are then joined to a single new node.

Example for $n = 4$



The cone $C_{45,3}$

Since we can apply a permutation $\sigma \in S_5$ on taxa, without loss of generality, we suppose that the first cherry to be picked is the cherry with leaves 4 and 5. This is true for all input vectors $\mathbf{d}$ which satisfy

$$(\mathbf{h_{10,i}}, \mathbf{d}) \geq \mathbf{0} \text{ for } \mathbf{i} = \mathbf{1}, \ldots, \mathbf{9},$$

where the vector

$$\mathbf{h_{ij}^{(n)}} := -\mathbf{A^{(n)}}(\mathbf{e_i} - \mathbf{e_j}).$$

Then, the set of all input vectors $\mathbf{d}$ for which the first picked cherry is 4-5 and the second one is 1-2:

$$C_{45,3} := \{\mathbf{d} \mid (\mathbf{h}_{10,i}, \mathbf{d}) \geq 0 \text{ for } i = 1, \ldots, 9, \text{ and } (\mathbf{r}_1 - \mathbf{r}_2, \mathbf{d}) \geq 0, (\mathbf{r}_1 - \mathbf{r}_3, \mathbf{d}) \geq 0\}$$

where $\mathbf{r}_1$, $\mathbf{r}_2$ and $\mathbf{r}_3$ are the first three rows of $-A^{(4)}R^{(5)}$.

The NJ cones

For $n = 5$, there is only one unlabeled tree and there are 15 labeled trees. There are 30 cones in the 5 dimension (i.e. there are two cones per a labeled tree).

- They do not form a fan.
- The union of cones $C_{12,3}$ and $C_{45,3}$ does not form a convex body (i.e. the union of two cones for one tree topology does not form a convex cone).

For $n = 6$

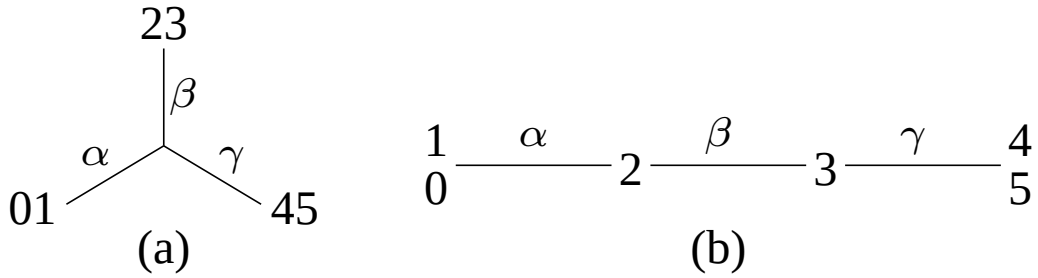


Figure 1: The two possible topologies for trees with six leaves, with edges connecting to leaves shrunk to zero.

There are three different classes of cones which cannot be mapped onto each other by the group action, C_I, C_{II}, C_{III}.

- **Type I**: $a, b, c, d, e, f \rightarrow a, b, c, d, (ef) \rightarrow a, b, (cd), (ef), \rightarrow$ Fig. 1(a)

- **Type II**: $a, b, c, d, e, f \rightarrow a, b, c, d, (ef) \rightarrow a, b, (cd), (ef)$
  $\rightarrow cd - a - b - ef$ (like Fig 1(b), but different labels)

- **Type III**: $a, b, c, d, e, f, \rightarrow a, b, c, d, (ef) \rightarrow a, b, c, (d(ef))$
  $\rightarrow ab - c - d - ef$ (exactly as in Fig 1(b))

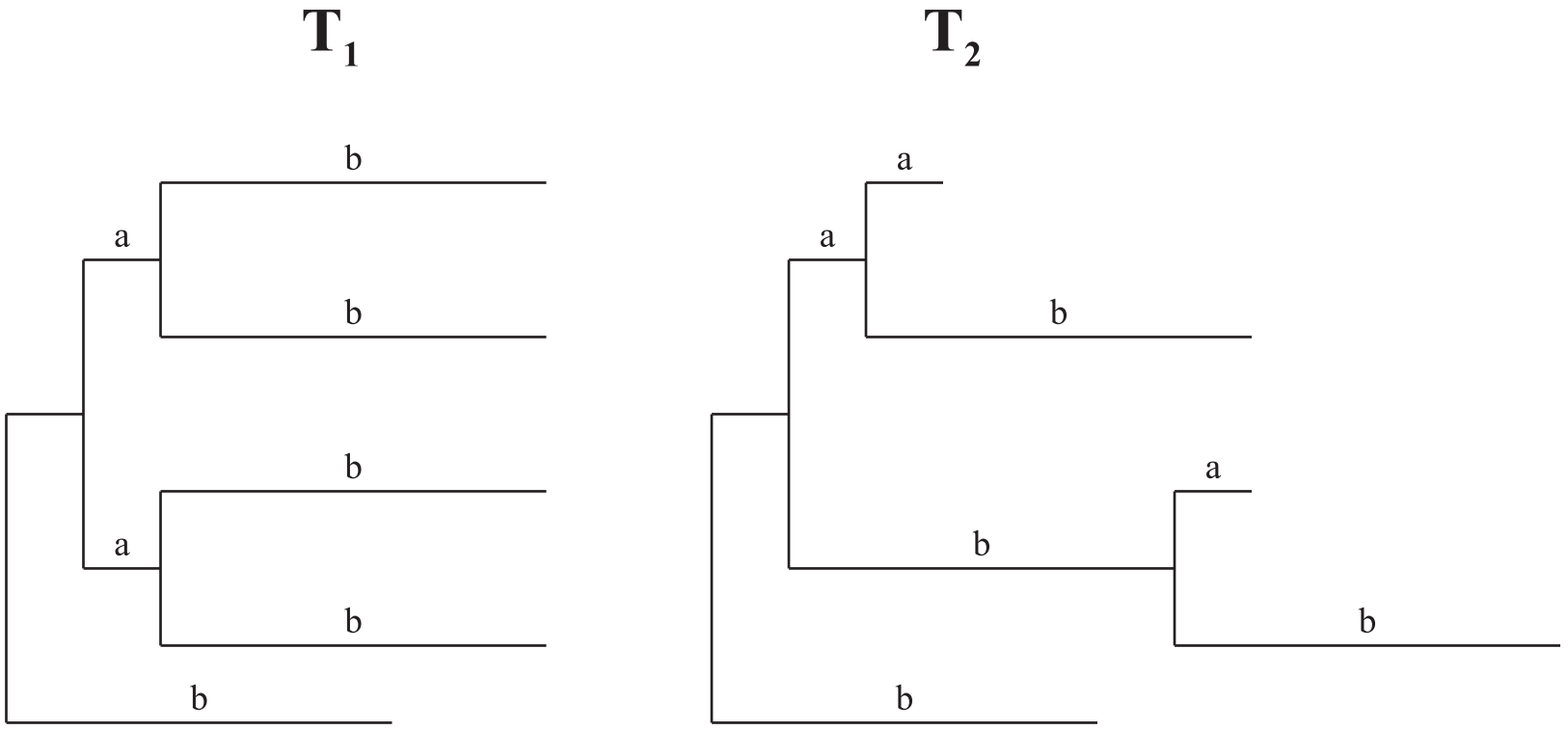
	C_{I}	C_{II}	C_{III}
stabilizer	$\langle (12), (34), (56) \rangle$	$\langle (12), (56) \rangle$	$\langle (12), (56) \rangle$
size of stabilizer	8	4	4
number of cones	90	180	180
cones giving same labeled topology	6	2	2

Simulation Results

With the Juke Cantor and Kimura 2 parameter models.

Consider two tree models...

Modeled from Strimmer and von Haeseler.



We generate $10,000$ replications at the edge length ratio, a/b $= 0.03/0.42$ for sequences of length 500BP with the Jukes-Cantor and Kimura 2 parameter models via a software `evolver` from `PAML` package.

For each set of 5 sequences, we compute first pairwise distances via the heuristic MLE method using a software `fastDNAml`. To compute cones, we used `MAPLE` and `polymake`.

We say an input vector (distance matrix) is **correctly classified** if the vector locates in one of the cones where the vector representation of the tree metric (noiseless input) lies. We say an input vector is **incorrectly classified** if the vector locates in the complement of the cones where the vector representation of the tree metric lies.

For distance matrices which are correctly classified by the NJ algorithm, we compute the minimum distance to any cone giving a different tree topology.

Distances of correctly classified vectors from closest misclassified vector

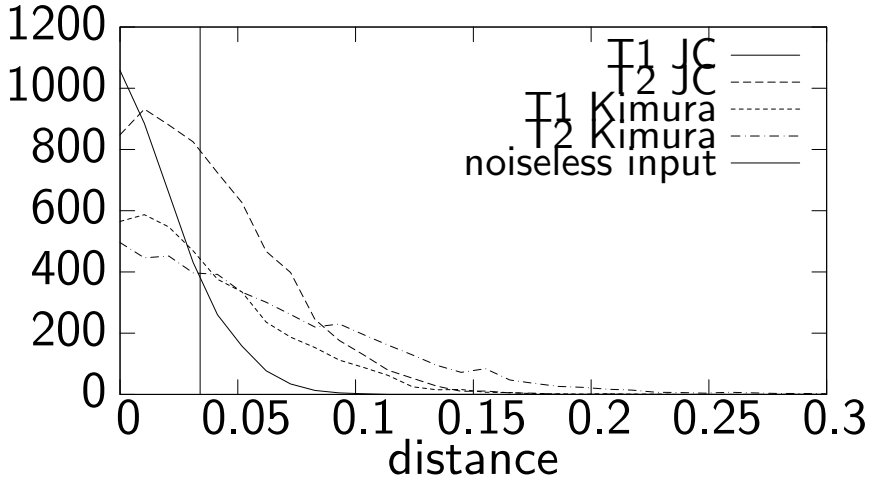


Figure 2: Distances of correctly classified input vectors from the closest correctly classified vector.

Mean and variance of the distances of correctly classified vectors from the nearest misclassified vector.

	JC		**Kimura2**	
	T1	T2	T1	T2
# of cases	3,581	6,441	3,795	4,467
Mean	0.0221	0.0421	0.0415	0.0629
Variance	$2.996 \cdot 10^{-4}$	$9.032 \cdot 10^{-4}$	$1.034 \cdot 10^{-3}$	$2.471 \cdot 10^{-3}$

For input vectors to which the NJ algorithm answers with a tree topology different from the correct tree topology, we compute the distances to the two cones for which the correct answer is given and take the minimum of the two. The bigger this distance is, the further we are off.

Distances of misclassified input vectors from closest correctly classified vector

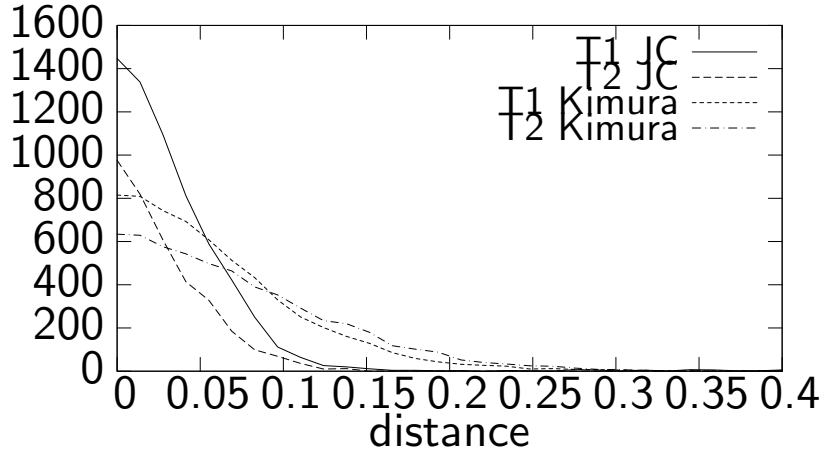


Figure 3: Distances of correctly incorrectly classified input vectors from the closest incorrectly classified vector.

Mean and variance of the distances of misclassified vectors to the nearest correctly classified vector.

	JC		**Kimura2**	
	T1	T2	T1	T2
# of cases	6,419	3,559	6,205	5,533
Mean	0.0594	0.0331	0.0951	0.0761
Variance	0.0203	$7.39 \cdot 10^{-4}$	0.0411	$3.481 \cdot 10^{-3}$

Future work

"On the optimality of the neighbor-joining algorithm" by K. Eickmeyer, P. Huggins, L. Pachter, R. Y.

In fact, the ME tree topology is also determined by polyhedral subdivisions of the spaces of dissimilarity maps $\mathbb{R}_+^{\binom{n}{2}}$. By comparing these two polyhedral subdivision, we study the optimality of the NJ algorithm.

In particular, we investigate and compare the polyhedral subdivisions for $n \leq 10$. (For $n = 4$, it is 100%, for $n = 5$, it is 99.5%, for $n = 6$, with the catapilar, 91.34%, with 3 cherries, 90.34 %, for $n = 7$, and it is about 78.87% with 2-cherry and 82.46% with 3-cherry etc.)

It will be submitted soon (hopefully).

Thank you....

The preprint is available at math.CO/0703081.