

Neighbor Joining

Algorithm: NJ

Input:

A matrix O of observed pairwise distances on k taxa.

Initialization:

Let $L = 1, \dots, k$ be the set of active nodes.

For all pairs of taxa, i, j , compute $\Delta[i, j]$ as defined in Equation 1.

Iteration:

While ($|L| > 2$)

{

Find a pair i and j in L such that $\Delta[i, j]$ is minimal.

Create a new node l with leaf nodes i and j .

Let the length of edge (l, i) be $D[l, i] = (D[i, j] + R_i - R_j)/2$.

Let the length of edge (l, j) be $D[l, j] = D[i, j] - D[l, i]$.

For all $m \neq i, j$,

set $D[l, m] = (D[i, m] + D[j, m] - D[i, j])/2$.

Remove i and j from L and add l

For all $l, m \in L$,

recompute R_l and $\Delta[l, m]$ as defined in Equation 1.

}

Termination:

When L consists of two nodes i and j ,

add the remaining edge between i and j , with length $D[i, j]$.

In order to compensate for differences in mutation rate, we define a corrected distance,

$$\Delta[i, j] = D[i, j] - R_i - R_j, \quad (1)$$

by subtracting the averaged distance to all other leaves,

$$R_i = \frac{1}{|L| - 2} \sum_{l \in L} D[i, l].$$

Note that Δ , which is used to determine which nodes are neighbors, differs from D , the matrix of pairwise tree distances. This is a modified version of the algorithm given in Durbin *et al.*