Neighbor-Joining Algorithm

Input

    We are given a set of samples $\mathcal{X}$ and a dissimilarity map δ on $\mathcal{X}$.

Initialization

- Create a star tree with center vertex c and an edge (c, u) between c and all samples $u \in \mathcal{X}$.

- Let N_c be the set of neighbors of c and $n = |N_c|$ (cardinality of N_c). Set d equal to δ.

Iterative Step

    While $n > 2$, use the following procedure to update the tree and find edge weights:

(a) Find vertices f, g that minimize the Q-criteria. Note that UPGMA would only use the first term in this formula, $d(i, j)$. The remaining terms represent how far i and j are from the other vertices.

$$Q(i, j) = (n - 2) \cdot d(i, j) - S_i - S_j, \quad \text{where}$$

$$S_i = \sum_{k \in N_c} d(i, k)$$

(b) Join f and g at internal vertex v. Now N_c contains v but not f and g. Compute the new edges weights:

$$d(f, v) = \frac{1}{2}d(f, g) + \frac{1}{2(n - 2)}[S_f - S_g]$$

$$d(g, v) = \frac{1}{2}d(f, g) + \frac{1}{2(n - 2)}[S_g - S_f]$$

(c) Compute the distances from v to all remaining vertices $i \in N_c$:

$$d(i, v) = \frac{1}{2}[d(f, i) - d(f, v)] + \frac{1}{2}[d(g, i) - d(g, v)]$$

Termination

    When $n = 3$, the tree topology does not change since we have obtained a binary tree. We still need to run the last iteration though to determine the 3 remaining edge weights. The output is then the tree topology and all edge weights.