Grade: 100

Algorithms in computational Biology - HW #5

Submit:

- Ilay Anais,
- Hadar Pur

Problem 1:

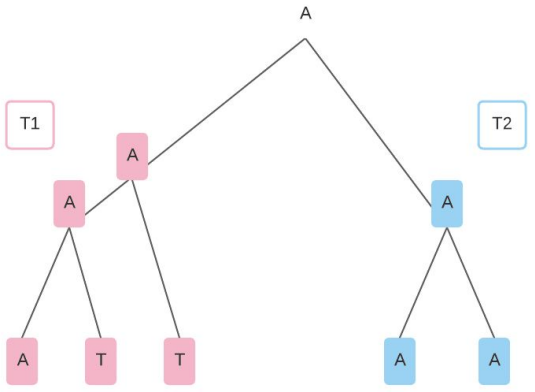
We proved in class (Lecture #10) a fundamental claim as basis for Fitch's algorithm for maximum parsimony. The claim considered a binary phylogenetic tree T with two principle subtrees T1 and T2 and states assigned to all leaves.

Claim a:

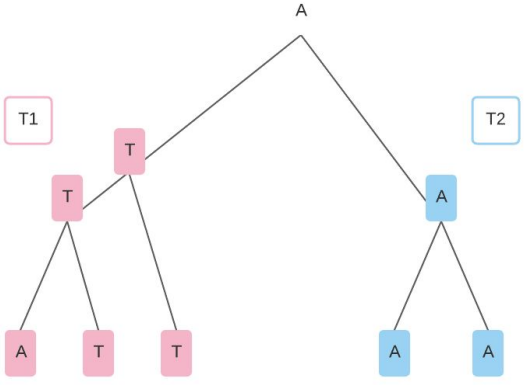
In all maximum parsimony assignments to nodes in T, the induced assignments to T1 and T2 are also optimal.

The claim is not true, we provide an explicit counter example:

In the following example, T1 is not optimal, yet an optimal parsimony still happens, as there are two mutations in the tree total.



In the example below T1 is optimal yet there are still two mutations in the tree, but both trees are optimal. Hence not all maximum parsimony assignments have an optimal T1 and T2.



Claim B:
In all maximum parsimony assignments to nodes in T, the induced assignment to either T1 or T2 (or both) is also optimal.

The claim is true, Proof:
Assume towards a contradiction that T is optimal, and neither T1 or T2 is optimal.
Let define:

- S1 be the score of T1 (the amount of mutations in T1)
- S2 be the score for T2 (the amount of mutations in T2)
- S be the score for T (the amount of mutations in T)

Then $S \geq S1 + S2$
Since neither T1 or T2 are optimal, there exist S' scores which are smaller than S1 and S2

As the scores go down by increments of integers

- $S1 \geq S1' + 1$
- $S2 \geq S2' + 1$

So we get that:
$S1' + S2' + 2 \leq S1 + S2 \implies S1' + S2' + 1 < S1 + S2$ which implies that $S' < S$.

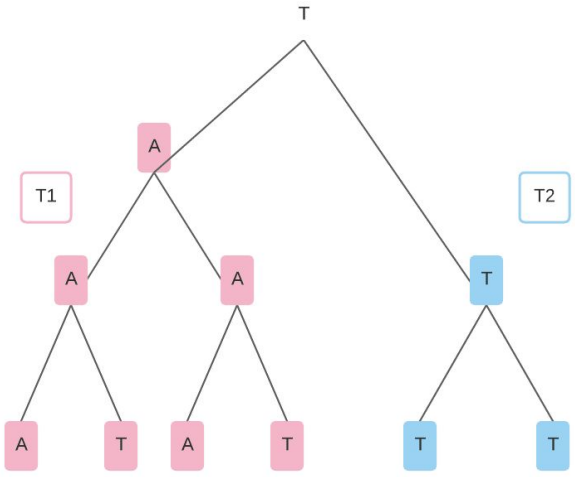
It means that there is a more optimal score than S, meaning there is a tree more optimal than T, which is a contradiction to our initial assumption.

Claim C:

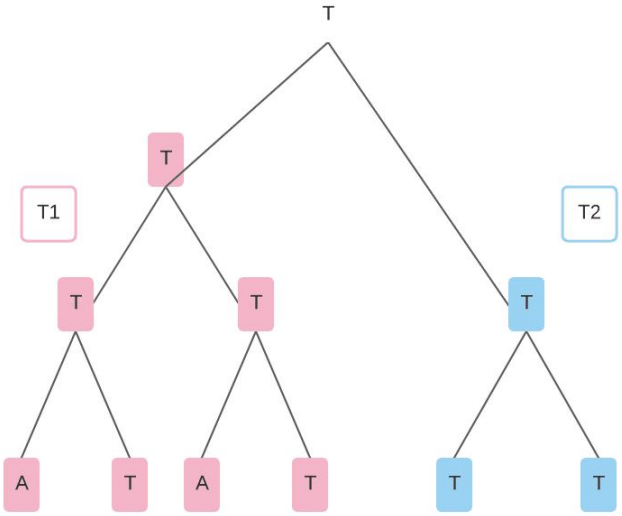
Every maximum parsimony assignment to T1 and T2 can be extended to an optimal assignment to T.

The claim is not true, we provide an explicit counter example:

In this first tree we see that T1 is optimal (has 2 mutations), and T2 is optimal (no mutations), yet we have 3 mutations in the tree T.



In the tree below, T1 is still optimal (with 2 mutations), and T2 is still optimal (no mutations), yet the tree T has 2 mutations total, meaning it is optimal, and more optimal then the tree above.



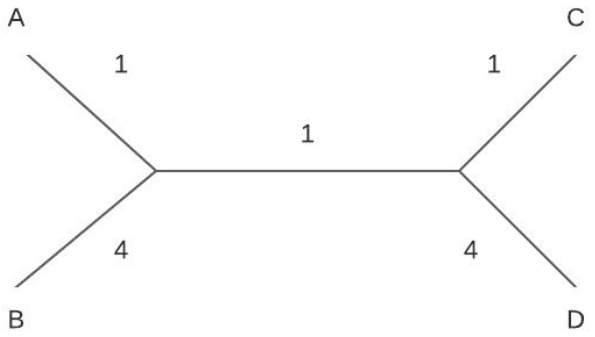
Problem 2:
The purpose of this problem is to examine consistency of the UPGMA algorithm for phylogeny reconstruction. The algorithm is mentioned in slides #60 in Lecture 11), and described in detail below:

Initialization: initialize n clusters for each of the n taxa.

Repeat for $\boldsymbol{n-1}$ times:

- Determine the cluster pair A, B which minimizes $d(A, B) = \frac{1}{|A||B|}\Sigma_{i\varepsilon A, j\varepsilon B} d(i, j)$
- Join the two clusters A, B by adding two edges that connect their roots.

Section A:
Topology tree proposed:



Distance matrix:

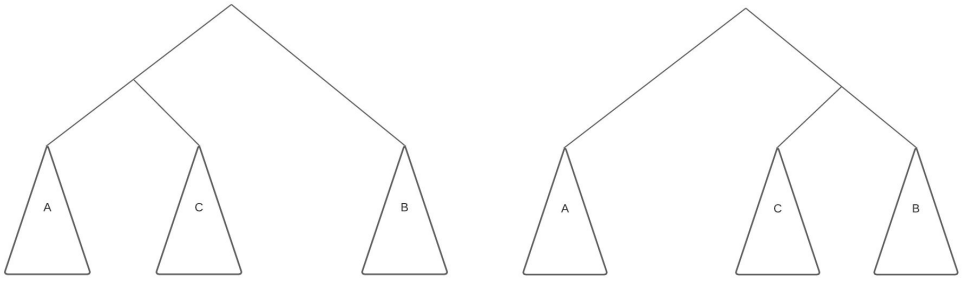
A	B	C	D	
0	1 + 4 = 5	1 + 1 +1 = 3	1 + 1 + 4 = 6	A
0	0	1 + 1 + 4 = 6	4 + 1 + 4 = 9	B
0	0	0	4 + 1 = 5	C
0	0	0	0	D

So the UPGMA algorithm will join A and C together as they have the shortest distance between them.
The resulting topology is (AC|BD), which is inconsistent with the tree we created at the top, as A and C are not neighbors.

Section B:

Claim:

If clusters A, B correspond to disjoint subtrees of T, and they are not neighbors (they are not sister subtrees), then there is a subtree C, which is disjoint to both of them and $\min\{d(A, C), d(B, C)\} < d(A, B)$.



Proof:

If clusters A, B correspond to disjoint subtrees of T, and they are not neighbors, then there are at least 2 edges between them, where a cluster v, which is disjoint to both of them must reside. Due to the tree being ultrametric, the distance between two neighbors cannot be larger than the distance between two non-neighbors, as was the case in the previous question.

This cluster v has to correspond to a subtree C which is a neighbor to one of them, and therefore is joined to either A or B through the algorithm.

We will prove that $\min\{d(A, C), d(B, C)\} < d(A, B)$.

UPGMA distance definition: $d(A, B) = \frac{1}{|A||B|}\Sigma_{i\varepsilon A, j\varepsilon B} d(i, j)$

Assume towards a contradiction that $\min\{d(A, C), d(B, C)\} > d(A, B)$ so it means that d(A, B) < d(A, C) **and** d(A, B) < d(B, C) .

We know that A and B are not neighbors, and subtree C must be between them, which means that either d(A, C) or d(B, C) or both are smaller than d(A, B).
Then it leads us to d(A, B) > d(A, C) or d(A, B) > d(B, C) in contradiction to our initial assumption.

As such, since this holds for the entire algorithm, and the tree T which we started with was ultrametric, the subtrees all are consistent with their distances from each other. So the T' constructed from this algorithm must have the same structure as the T from which the distance matrix came from.