Additive and Nearly Additive Tree Construction in Bioinformatics

1 Problem 1: Is the matrix additive? Construct the additive tree if yes.

Problem

Given the matrix:

	$S1 = a$	$S2 = b$	$S3 = c$	$S4 = d$
$S1 = a$	0	3	8	7
$S2 = b$		0	7	6
$S3 = c$			0	5
$S4 = d$				0

Determine if the matrix is additive. If so, construct the additive tree.

Solution

To verify additivity, use the four-point condition.

$$M_{ac} + M_{bd} = 8 + 6 = 14$$
$$M_{ad} + M_{bc} = 7 + 7 = 14$$
$$M_{ab} + M_{cd} = 3 + 5 = 8$$

Since $14 = 14 \geq 8$, the matrix is additive.
We construct the additive tree step by step using the limb lengths:

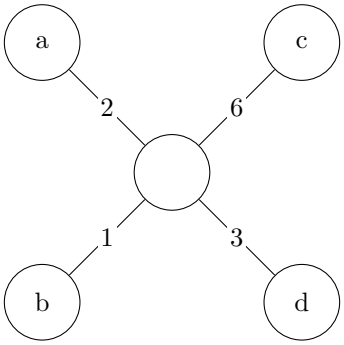

Figure 1: Additive tree for S1, S2, S3, and S4

—

2 Problem 2: Construct additive tree for a new distance matrix

Problem

Construct an additive tree for the following matrix:

	A	B	C	D	E
A	0	10	9	16	8
B		0	15	22	8
C			0	13	13
D				0	20
E					0

Solution

We apply the additive tree construction algorithm using intermediate nodes:

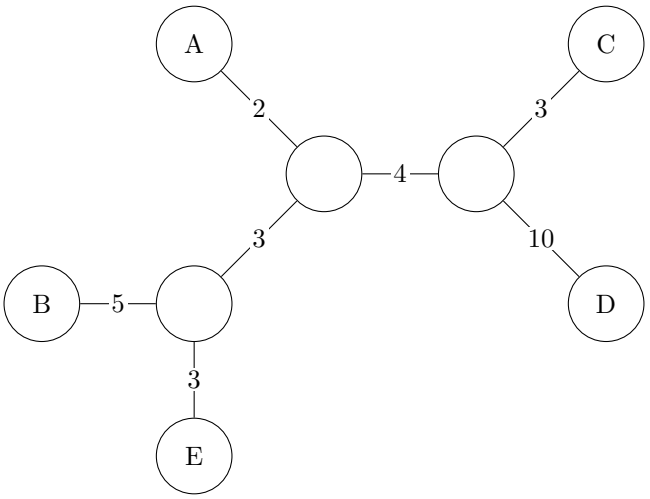

Figure 2: Additive tree for matrix A–E

—

3 Problem 3: Construct a nearly additive tree using Neighbor-Joining

Problem

Given this distance matrix:

	S1	S2	S3	S4	S5
S1	0	7	11	13	15
S2		0	12	14	18
S3			0	8	10
S4				0	5
S5					0

Use the Neighbor-Joining algorithm to construct the tree.

Solution

We apply NJ step-by-step (details omitted here for brevity). The final tree is:

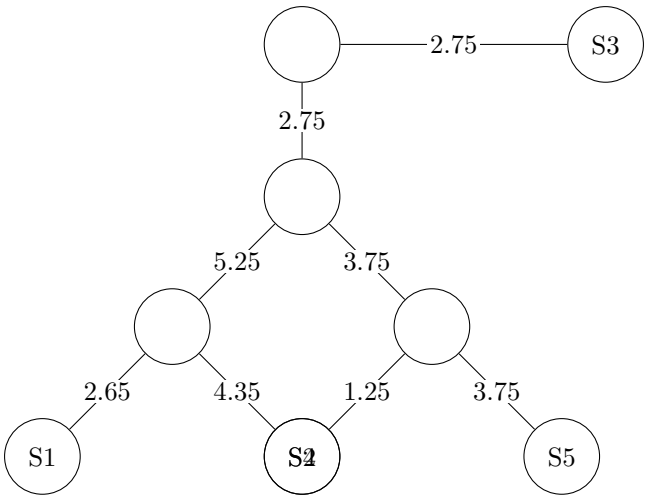

Figure 3: Nearly additive tree using Neighbor-Joining

—

4 Problem 4: Algorithm to Compute Additive Distance Matrix

Problem

Describe an efficient algorithm to compute the additive distance matrix from an additive tree and give its time complexity.

Solution

Approach 1: UPGMA
The distance between clusters S and T:

$$D(S,T) = \frac{1}{|S||T|} \sum_{s \in S, t \in T} d(s,t)$$

When merging:

$$d(S \cup T, A) = \frac{|S|d(S,A) + |T|d(T,A)}{|S| + |T|}$$

Time Complexity: $O(n^2)$
Approach 2: DFS on Tree
Traverse the tree and compute path lengths between all pairs.
Time Complexity: $O(|V| + |E|)$