1. Phylogenetic analysis

Slides for the Lecture on
Foundations of Sequence Analysis
Winter 2022/2023
Stefan Kurtz

January 31, 2023

- Given a collection of species, the goal of phylogenetic analysis is to determine and describe the evolutionary relationship between the species.
- In particular, this involves determining the order of speciation events and their approximate timing.
- It is generally assumed that speciation is a branching process: a population of organisms becomes separated into two sub-populations.
- Over time, these evolve into separate species that do not cross-breed.
- Because of this assumption, a tree is often used to represent a proposed phylogeny for a set of species, showing how the species evolved from a common ancestor.

Phylogenetic trees

- In the following, we will use X to denote a finite set of taxa.
- A taxon $x \in X$ is simply a representative of a group of individuals defined in some way.

A phylogenetic tree (on X) is a triple $T = (V, E, \lambda)$ consisting of a connected graph (V, E) without cycles, together with a one-to-one labeling λ of the leaves by elements of X. There are two kinds of phylogenetic trees*:

1. Rooted trees reflect the most basal ancestor of the tree in question.
2. Unrooted trees do not imply a known ancestral root.

https://www.ncbi.nlm.nih.gov/Class/NAWBIS/Modules/Phylogenetics/phylo9.html

Phylogenetic trees

An unrooted phylogenetic tree for 9 taxa placed on the leaves. All internal nodes have degree 3. Such a tree is often displayed in a circular layout.

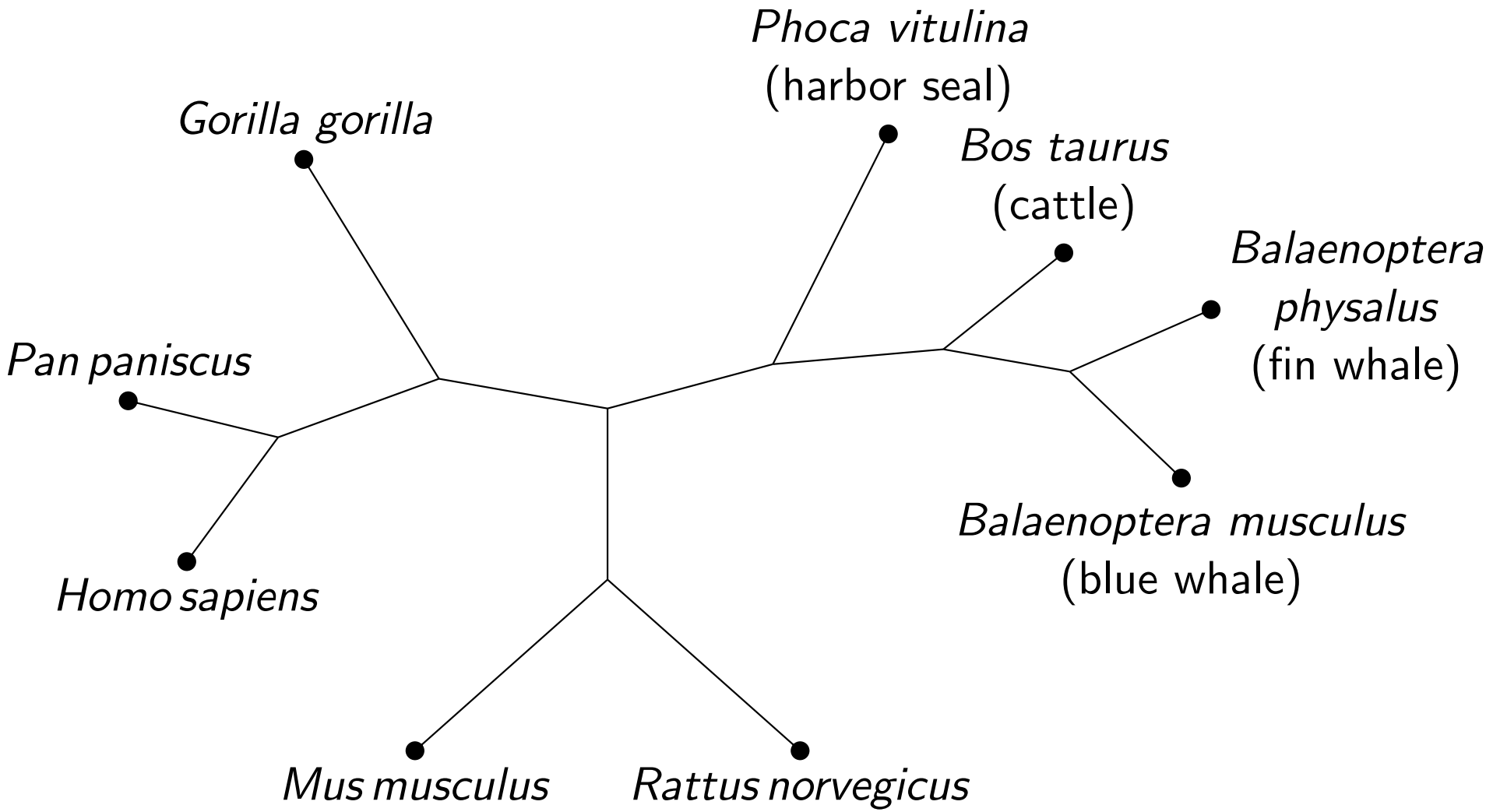


Figure 1: A rooted phylogenetic tree of life, derived from the comparison of rRNA genes. It shows the three major kingdoms Bacteria, Archaea, and Eucaryota.

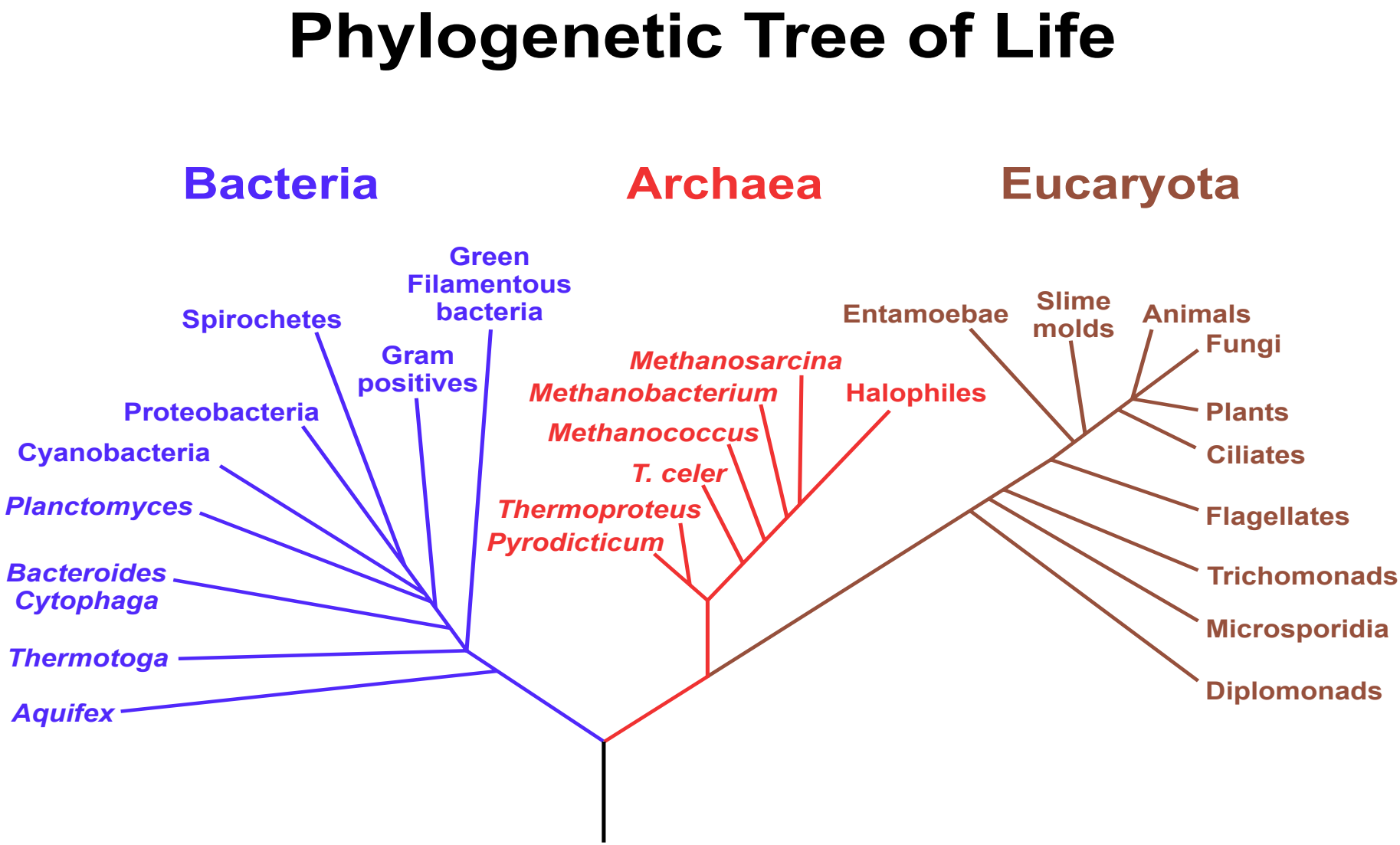


Figure 2: Phylogenetic tree of novel coronavirus according to `https://nextstrain.org/ncov/global`.

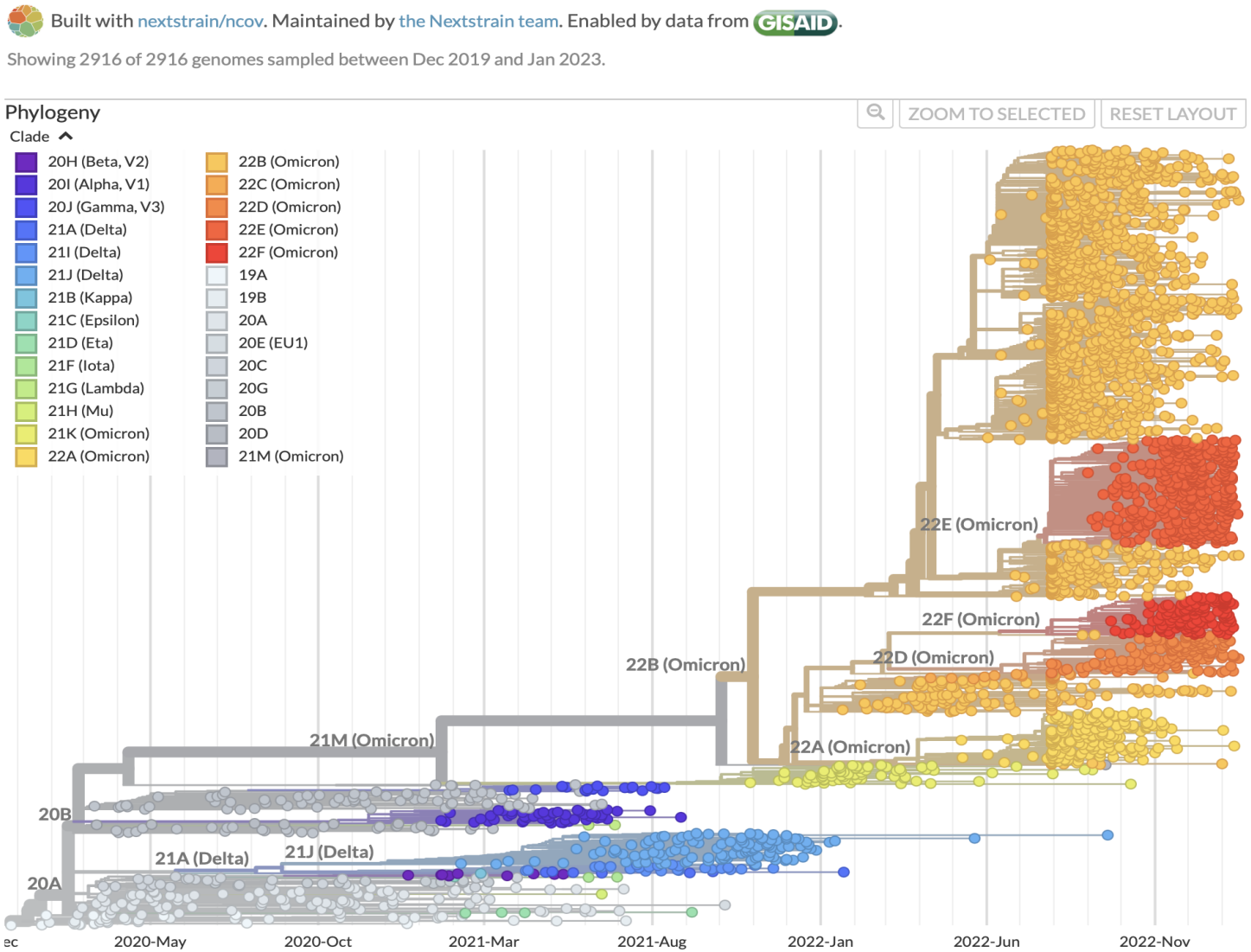
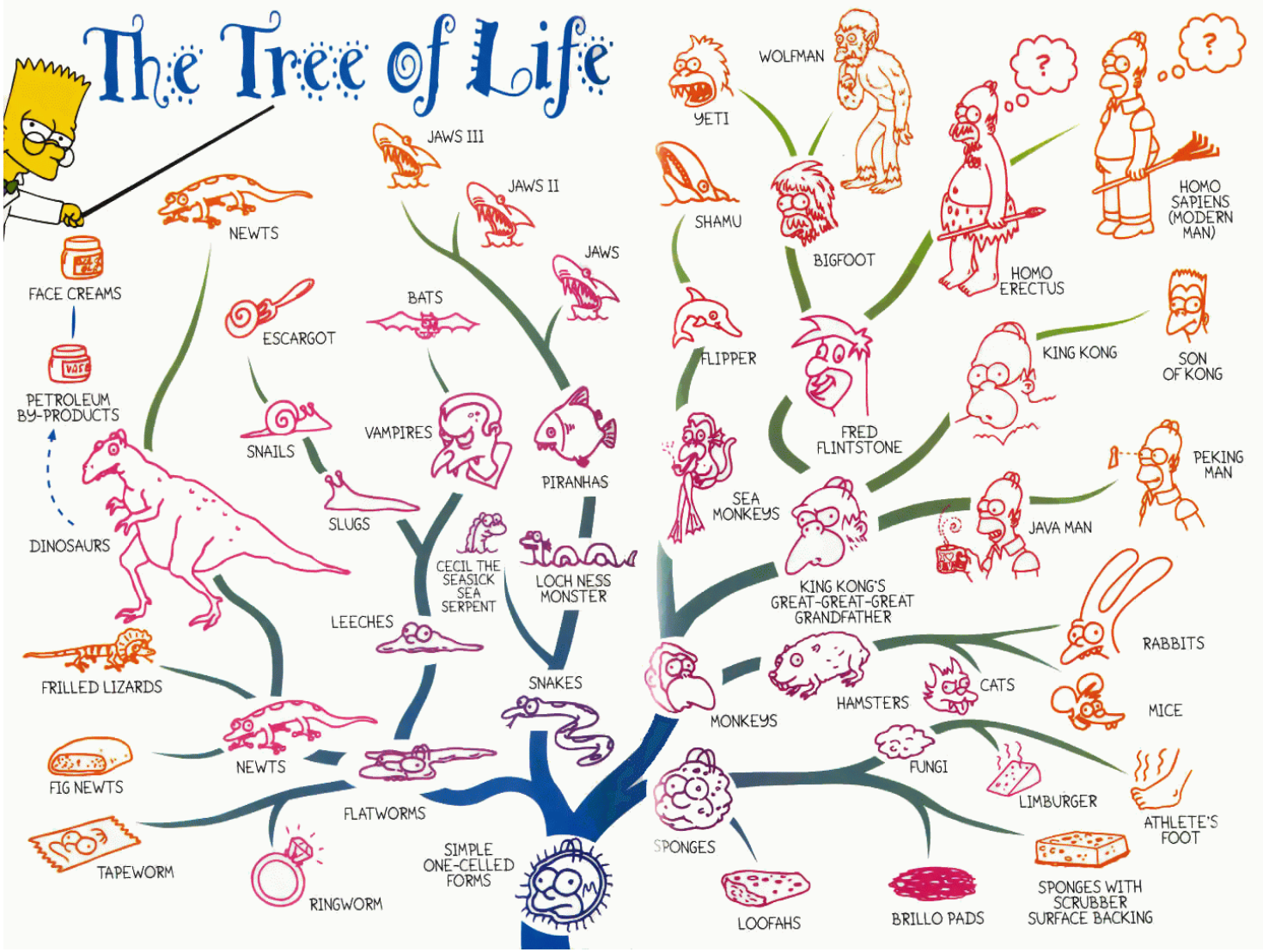


Figure 3: A modern version of the (rooted) tree of life, according to Homer Simpson, `https://www.pinterest.com/pin/526639750148386179/`



Edge lengths

- A phylogenetic tree describes the putative order of speciation events that gave rise to the considered taxa.
- Additionally, one assigns positive length to each edge of the tree.
- Ideally, these lengths should be proportional to the amount of time that lies between speciation events.
- However, in practice the edge lengths usually represent quantities obtained by some given computation and only correspond very indirectly to time.

Approaches to construct phylogenetic trees

There are three main approaches to constructing phylogenetic trees from molecular data.

1. *Distance methods* first compute a *distance function* from a given set of biological data and then determine a tree representing these distances as closely as possible.
2. *Maximum parsimony* takes as input a set of aligned sequences and attempts to find a tree and a labeling of its internal nodes by auxiliary sequences such that the number of mutations along the edges of the tree is a minimum.
3. Given a probabilistic model of evolution, *maximum likelihood approaches* aim at finding a phylogenetic tree that maximizes the likelihood of obtaining the given sequences.

Here we will focus on distance methods. For a set X of taxa we assume a distance function $d : X \times X \to \mathbb{R}^0_+$ that associates a distance $d(x, y)$ with every pair of taxa $x, y \in X$. We usually require that d is a metric.

Additivity and the four-point condition

We next consider an important notion relating distance functions and phylogenetic trees.

Definition 1

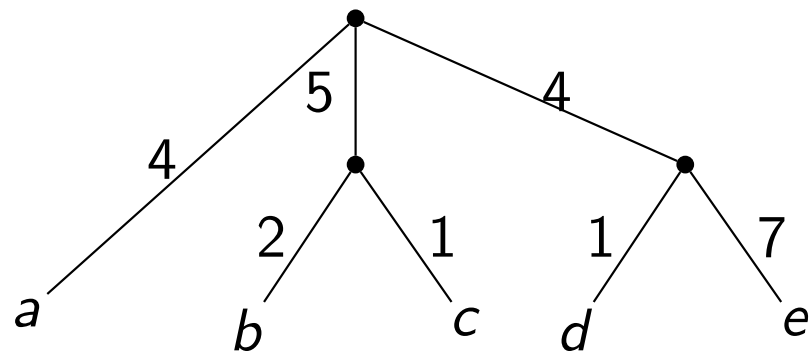
- Let X be a set of taxa and T_X be a phylogenetic tree over X.
- Let $d : X \times X \to \mathbb{R}^0_+$ be a distance function.
- d is *additive* with respect to T_X, if it was obtained by adding up edge lengths on paths between the leaves of T_X.
- That is, for all x, $y \in X$, $d(x, y)$ is the sum of the edge lengths of the unique path in T_X from x to y.

In the following we abbreviate 'with respect to' by 'w.r.t.'.

Example 1

Let $X = \{a, b, c, d, e\}$. Here is an example of a distance function $d : X \times X \to \mathbb{R}^0_+$ which is additive w.r.t. to the phylogenetic tree on the right:

	a	b	c	d	e
a	0	11	10	9	15
b	11	0	3	12	18
c	10	3	0	11	17
d	9	12	11	0	8
d	15	18	17	8	0



For example, $d(a, e) = 4 + 4 + 7 = 15$ and $d(a, c) = 4 + 5 + 1 = 10$.

Definition 2

A distance function $d : X \times X \to \mathbb{R}^0_+$ is *additive*, if there is a phylogenetic tree T_X such that d is additive with respect to T_X.

Additivity and the four-point condition

Interestingly, we can decide whether a distance function is additive without determining an appropriate phylogenetic tree. The method is based on a result by Buneman.[1]

Theorem 3

A distance function d on X is additive, if and only if for any four (not necessarily distinct) elements $w, x, y, z \in X$ the so-called four-point condition holds:

$$d(w,x) + d(y,z) \leq \max\{d(w,y) + d(x,z), d(w,z) + d(x,y)\} \qquad (1)$$

Using this theorem, one can decide whether or not a distance function is additive. If $|X| = n$, then one has to enumerate all n^4 4-tuples $(w,x,y,z) \in X \times X \times X \times X$ of taxa and check whether condition (1) holds. This can be done in constant time for each 4-tuple. So the running time of such a decision method is $O(n^4)$.

[1] Buneman, *J. of Combinatorial Theory*, 1974

Example 2

Consider the distance function d for $X = \{A, B, C, D\}$, given by the following distance matrix:

	A	B	C	D
A	0	7	6	5
B	7	0	3	6
C	6	3	0	5
D	5	6	5	0

We evaluate

$d(A, B) + d(C, D) = 7 + 5 = 12$

$d(A, C) + d(B, D) = 6 + 6 = 12$

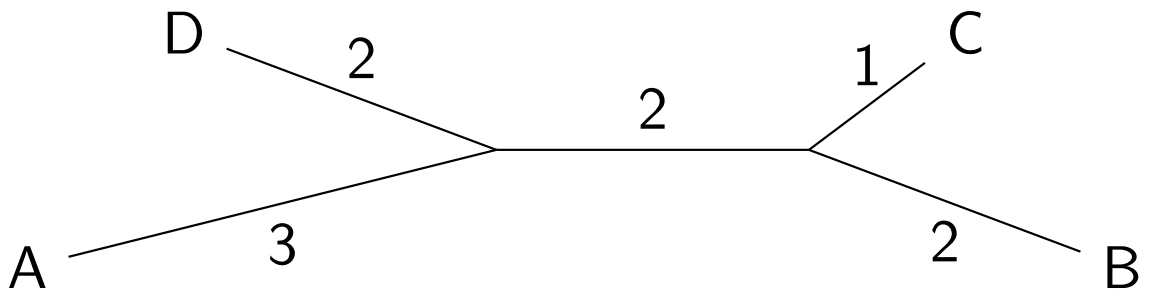
$d(A, D) + d(B, C) = 5 + 3 = 8$

and have to check $4^4 = 256$ 4-tuples.

For example, we verify

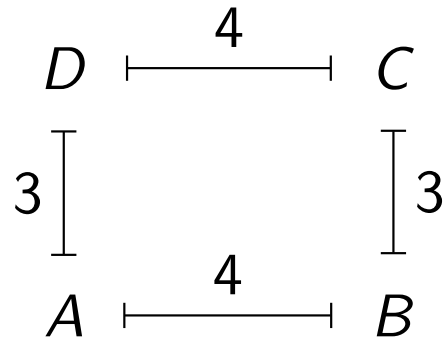
$$12 = d(A,B) + d(C,D) \leq \max\{d(A,C) + d(B,D), d(A,D) + d(B,C)\} = 12$$

Indeed, for all 4-tuples, the four-point condition holds. So d is additive. Here is a phylogenetic tree T_X such that d is additive w.r.t. T_X.



Example 3

Consider as distance function d the euclidean distances given in the plane:



The distance of D and B and of A and C is $\sqrt{4^2+3^2}=5$. We evaluate

$$d(A,B)+d(C,D)=4+4=8$$
$$d(A,C)+d(B,D)=5+5=10$$
$$d(A,D)+d(B,C)=3+3=6$$

Now $d(A,C)+d(B,D)=10>8=\max\{8,6\}=\max\{d(A,B)+d(C,D),$
$d(A,D)+d(B,C)\}$

Hence the four-point condition does not hold. So d is not additive.

The UPGMA reconstruction method

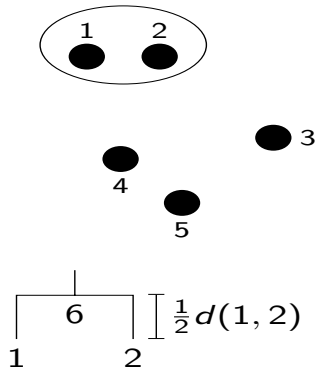
We will now present a simple distance method called UPGMA which stands for *unweighted pair group method using arithmetic averages*. It was first described by Sokal & Michener.[2]

- Given a set of taxa X and a distance function d, UPGMA produces a rooted binary phylogenetic tree T_X with edge lengths.
- It operates by clustering the given taxa, at each stage merging two clusters, and at the same time creating a new node in the tree.
- The tree is assembled "upwards", first clustering pairs of leaves, then pairs of clustered leaves etc.
- Each node is given a height and the edge lengths are obtained as the difference of heights of its two end nodes.

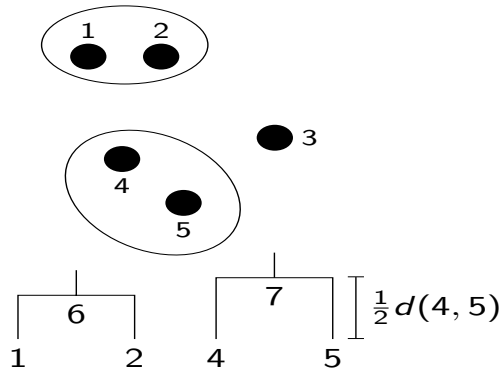
[2] Sokal & Michener, *University of Kansas Science Bulletin*, 1958

Example 4

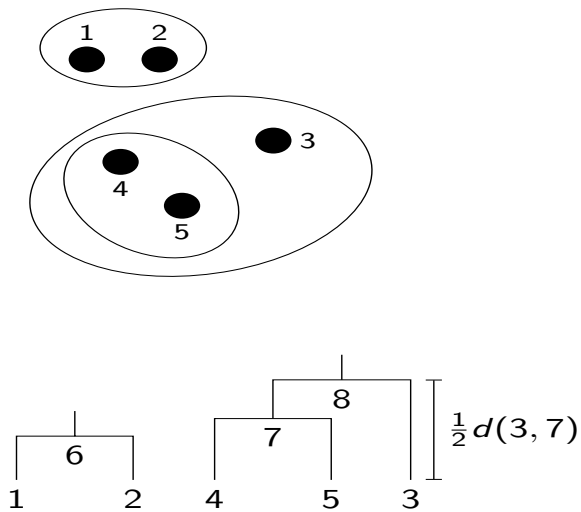
Let $X = \{1, 2, 3, 4, 5\}$ and consider distances in the plane. The different steps of the UPGMA method are shown in the following illustration:



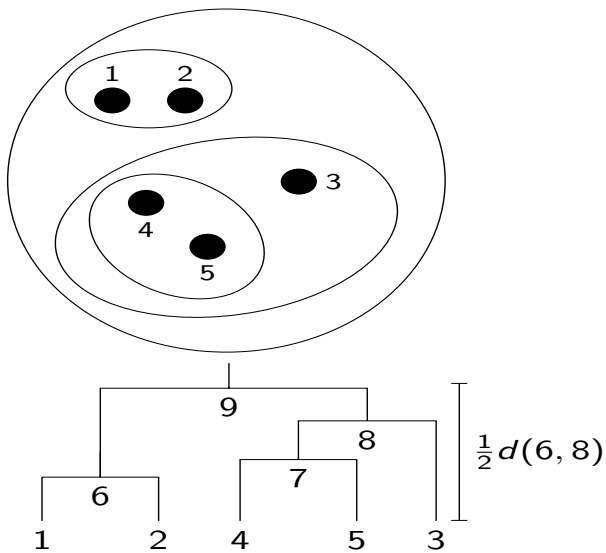
union of 1 and 2 $\Rightarrow$ cluster 6



union of 4 and 5 $\Rightarrow$ cluster 7



union of 7 and 3 $\Rightarrow$ cluster 8



union of 6 and 8 $\Rightarrow$ cluster 9

The UPGMA reconstruction method

- Initially, we are given a distance $d(x, y)$ between any two $x, y \in X$.
- We define the distance $d(C_i, C_j)$ between two clusters $C_i \subseteq X$ and $C_j \subseteq X$ with $C_i \cap C_j = \emptyset$ to be the average distance between all pairs of taxa from each cluster:

$$d(C_i, C_j) = \frac{1}{|C_i||C_j|} \sum_{x \in C_i} \sum_{y \in C_j} d(x, y)$$

- Obviously, $|C_i||C_j|$ is the number of pairs we can construct from the clusters C_i and C_j.
- We have $d(\{x\}, \{y\}) = d(x, y)$ according to this definition.
- The following Lemma shows how to efficiently compute the distance between two clusters.

The UPGMA reconstruction method

Lemma 1

If C_k is the union of two clusters C_i and C_j, and C_ℓ is any other cluster, then

$$d(C_k, C_\ell) = \frac{d(C_i, C_\ell)|C_i| + d(C_j, C_\ell)|C_j|}{|C_i| + |C_j|}$$

Proof.

$$
\begin{aligned}
d(C_k, C_\ell) &= \frac{1}{|C_k||C_\ell|} \sum_{x \in C_k} \sum_{y \in C_\ell} d(x,y) \\
&= \frac{\frac{1}{|C_\ell|} \left(\sum_{x \in C_i \cup C_j} \sum_{y \in C_\ell} d(x,y) \right)}{|C_k|} \\
&= \frac{\frac{1}{|C_\ell|} \left(\sum_{x \in C_i} \sum_{y \in C_\ell} d(x,y) \right) + \frac{1}{|C_\ell|} \left(\sum_{x \in C_j} \sum_{y \in C_\ell} d(x,y) \right)}{|C_i| + |C_j|} \\
&= \frac{|C_i| \frac{1}{|C_i| \cdot |C_\ell|} \left(\sum_{x \in C_i} \sum_{y \in C_\ell} d(x,y) \right) + |C_j| \frac{1}{|C_j||C_\ell|} \left(\sum_{x \in C_j} \sum_{y \in C_\ell} d(x,y) \right)}{|C_i| + |C_j|} \\
&= \frac{|C_i| d(C_i, C_\ell) + |C_j| d(C_j, C_\ell)}{|C_i| + |C_j|} \\
&= \frac{d(C_i, C_\ell)|C_i| + d(C_j, C_\ell)|C_j|}{|C_i| + |C_j|}
\end{aligned}
$$

Algorithm 1 (UPGMA reconstruction method)

Input: A set of taxa X and a corresponding distance function d.
Output: A binary, rooted phylogenetic UPGMA tree T.

Initialization: Assign each taxon x_i to its own cluster C_i. Define one leaf of T for each taxon, placed at height zero

Iteration:
- Determine a pair (C_i, C_j) of distinct clusters for which $d(C_i, C_j)$ is minimal.
- Define a new cluster k by $C_k = C_i \cup C_j$.
- Determine $d(C_k, C_\ell)$ in constant time for all existing clusters C_ℓ using the update formula from Lemma 1.
- Define a node k with child nodes i and j, and place it at height $\frac{1}{2}d(C_i, C_j)$.
- Add C_k to the current set of clusters; remove C_i and C_j.

Termination: When only 2 clusters C_i and C_j remain, place the root at height $\frac{1}{2}d(C_i, C_j)$ and connect it to its children i and j

Example 5

We consider a distance function over 5 sequences (*Bsu*, *Bst*, *Lvi*, *Amo*, and *Mlu*). The meaning of these abbreviations and a distance function (given as a matrix) is as follows:

Bsu *Bacillus subtilis*
Bst *Bacillus stearothermophilus*
Lvi *Lactobacillus viridescens*
Amo *Acholeplasma modicum*
Mlu *Micrococcus luteus*

	Bsu	*Bst*	*Lvi*	*Amo*	*Mlu*
Bsu	0.0	0.1715	0.2147	0.3091	0.2326
Bst		0.0	0.2991	0.3399	0.2058
Lvi			0.0	0.2795	0.3943
Amo				0.0	0.4289
Mlu					0.0

Each step of the UPGMA algorithm reduces the number of taxa by one. Hence the distance matrices become smaller and smaller. The first clustering step identifies *Bsu* and *Bst* to be the sequences with minimum distance 0.1715. These sequences are clustered, the new cluster is placed on a node at height $0.5 \cdot 0.1715 = 0.08575$ and the following distance matrix is produced:

Example 5

	$\{Bsu, Bst\}$	Lvi	Amo	Mlu
$\{Bsu, Bst\}$	0.0	0.2569	0.3245	0.2192
Lvi		0.0	0.2795	0.3943
Amo			0.0	0.4289
Mlu				0.0

Now the algorithm clusters $\{Bsu, Bst\}$ and Mlu, since their distance 0.2192 is minimal. This step creates a new cluster at a node of height $0.5 \cdot 0.2192 = 0.1096$ and leads to the following distance function:

Example 5

	$\{Bsu, Bst, Mlu\}$	Lvi	Amo
$\{Bsu, Bst, Mlu\}$	0.0	0.3027	0.3593
Lvi		0.0	0.2795
Amo			0.0

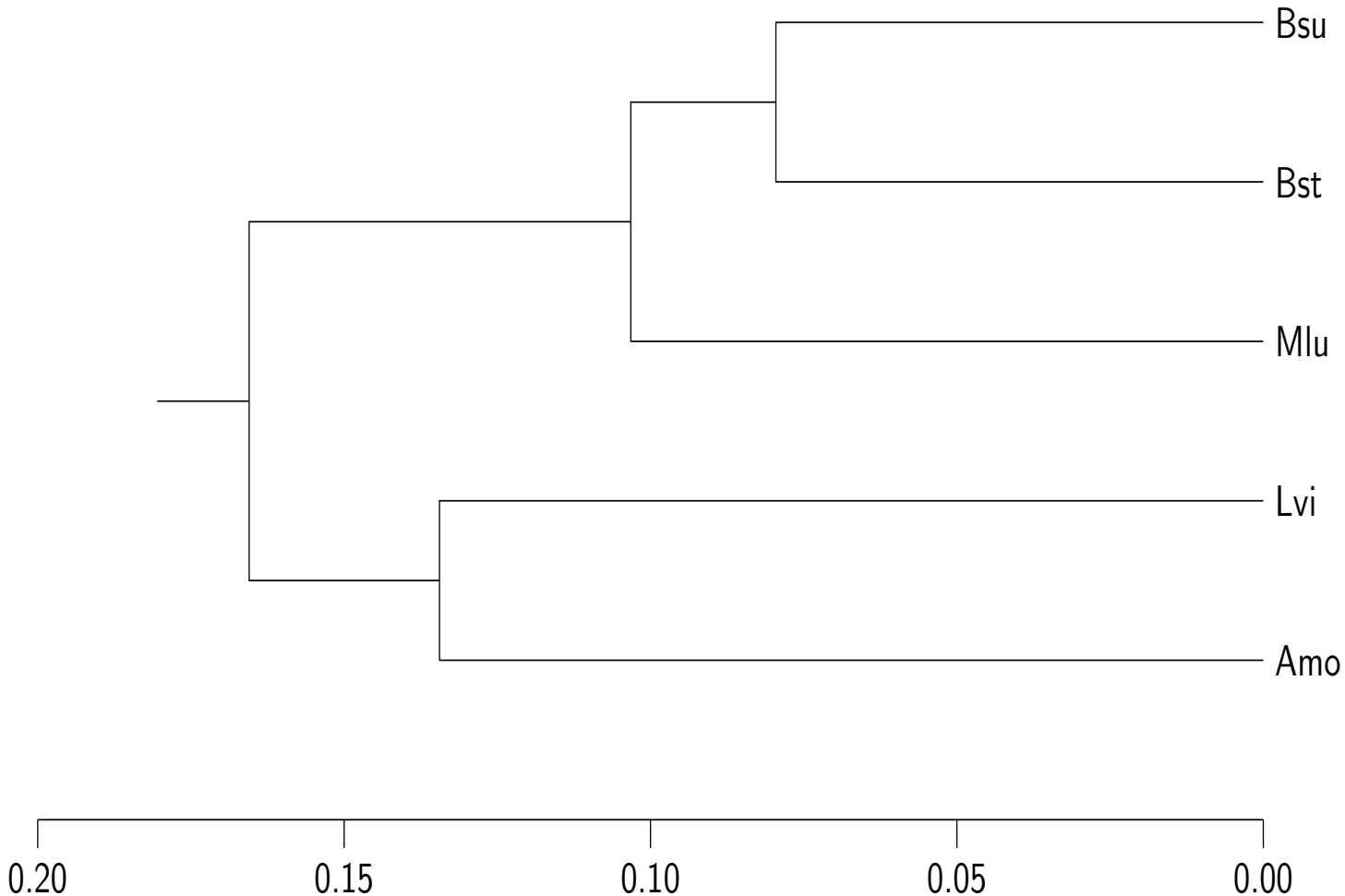
Now Lvi and Amo are clustered since their distance 0.2795 is minimal. This step creates a new cluster at a node of height $0.5 \cdot 0.2795 = 0.13975$ and the following distance matrix is produced:

Example 5

	$\{Bsu, Bst, Mlu\}$	$\{Lvi, Amo\}$
$\{Bsu, Bst, Mlu\}$	0.0	0.3310
$\{Lvi, Amo\}$		0.0

The method produces the following tree with a root at height $0.5 \cdot 0.3310 = 0.1655$:

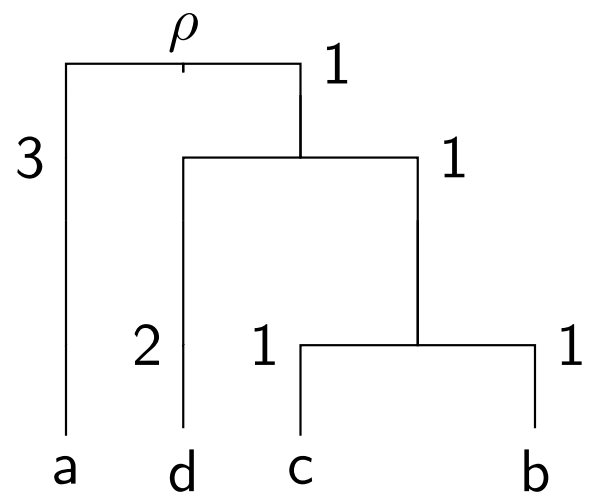
It is biologically incorrect, as we will see later.



Given a distance function $d : X \times X \to \mathbb{R}^0_+$, the UPGMA method aims at building a rooted tree T_X with the property that all leaves have the same distance from the root ρ, see the following example:

Example 6

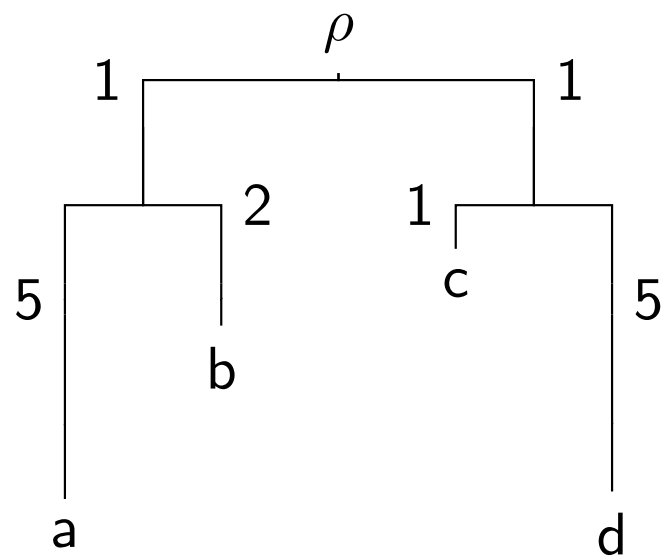
Here is a tree in which all leaves have the same distance 3 from the root.



The molecular clock hypothesis

- So the UPGMA method is suitable for distance functions based on sequence data that has evolved with a mutation rate that is constant over time.
- The hypothesis that evolutionary events happen at a constant rate is called the *molecular clock hypothesis*.
- If the distance function d on X is additive w.r.t. a binary rooted phylogenetic tree T_X that adheres to the molecular clock hypothesis, then the UPGMA method will construct this tree.
- Otherwise, if T_X does not adhere to the molecular clock hypothesis, then UPGMA may fail to reconstruct the tree correctly.
- For example, it cannot reconstruct the tree of Figure 4.

Figure 4: A tree that cannot be reproduced by the UPGMA algorithm, as e.g. $d(\rho, b) = 3 \neq 6 = d(\rho, a)$.



The ultrametric property

- Before one applies the UPGMA algorithm to a given distance function, one needs to know if the distance function is additive w.r.t. a phylogenetic tree in which all leaves are at the same distance from the root.
- Here the ultrametric property is of particular relevance.

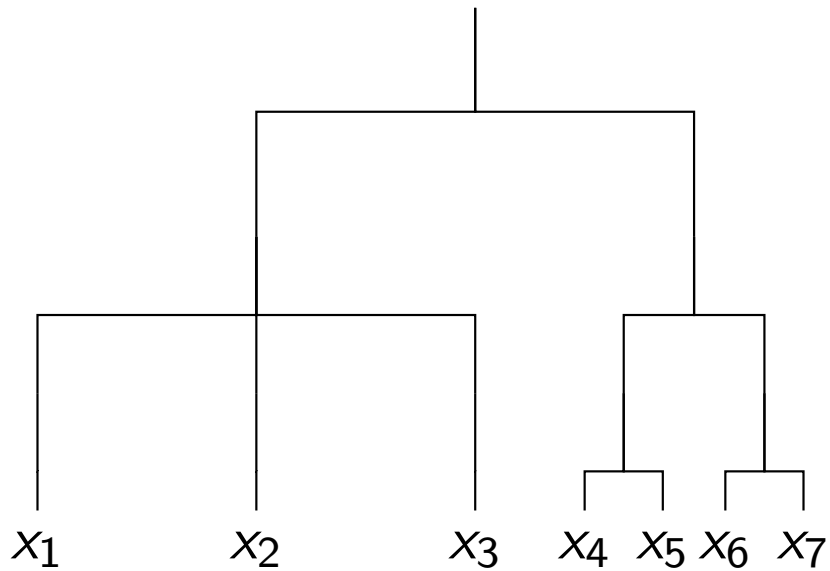
Definition 4

A distance function $d : X \times X \to \mathbb{R}_+^0$ is called an *ultrametric*, if for any triplet $(x, y, z) \in X \times X \times X$, the three distances $d(x, y)$, $d(x, z)$, and $d(y, z)$ have one of the following two properties:

1. all three distances are equal
2. two are equal and the remaining one is smaller

See also Figure 5, explaining the two ultrametric conditions 1 and 2.

Figure 5: A schematic phylogenetic tree, in which x_1, x_2, and x_3 satisfy condition 1 of Definition 4, and x_5, x_6, and x_7 satisfy condition 2 with $d(x_6, x_7) < d(x_5, x_6) = d(x_5, x_7)$.



The ultrametric property

- Note that one can decide whether or not a distance function is ultrametric, according to Definition 4.
- If $|X| = n$, then one has to enumerate all n^3 triplets (x, y, z) of taxa and check for each of these whether condition 1 and 2 holds.
- This can be done in constant time.
- Hence the decision requires $O(n^3)$ time.
- The next theorem relates the different notions introduced so far:

Theorem 5

Let $d : X \times X \to \mathbb{R}^0_+$ be a distance function that is additive w.r.t. tree T_X. Then d is ultrametric, if and only if every leaf in T_X has the same distance from the root. $\square$

The ultrametric property

- From the theorem we conclude that if d is additive w.r.t. some tree T_X and d is ultrametric, then every leaf in T_X has the same distance from the root.
- This implies that T_X can be reconstructed by the UPGMA-method.
- So it makes sense to check the additivity and the ultrametric property of the distance function, before applying the UPGMA-method.
- If both properties hold, the UPGMA will reconstruct the phylogenetic tree.

Neighbor-Joining

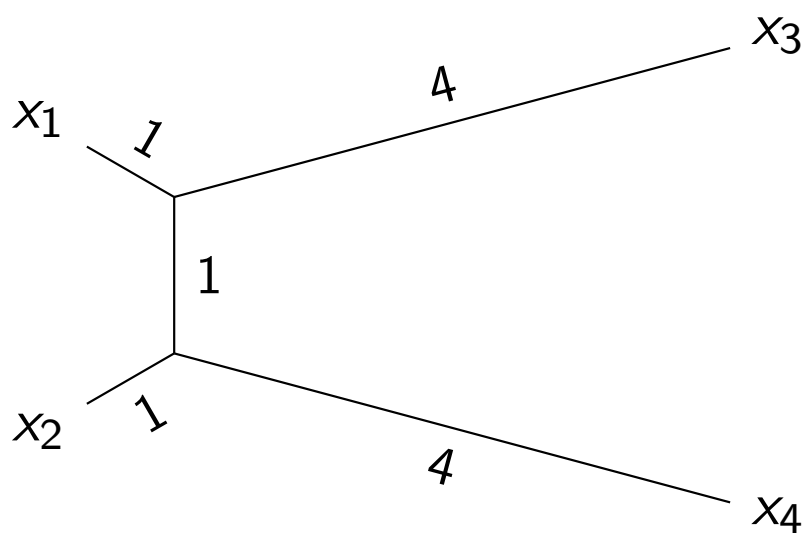
The most widely used distance based method to construct phylogenetic trees is the neighbor-joining method.[3]

- Given an additive distance function d, neighbor-joining produces an unrooted phylogenetic tree T with edge lengths.
- It is especially suitable, when the rate of evolution of the separate subtree under consideration varies.
- First, consider a tree T and let the distance function d be additive w.r.t. the leaves of T, i.e. d is a distance function defined on the leaves of T, obtained by adding edge lengths.
- The neighbor-joining method is based on the fact that we can decide which nodes are neighboring without knowing the tree, but only using the distance function.
- However, it does not suffice simply to pick the two closest taxa, i.e. a pair (i, j) with $d(i, j)$ minimal, see Figure 6.

[3] Saitou and Nei, *Mol. Biol. Evol.*, 1987

Figure 6: Suppose $X = \{x_1, x_2, x_3, x_4\}$ and distances derived from the tree below. Leaves x_1 and x_2 have minimum distance 3, but they are not neighbors in the tree. So the reconstructed tree based on first choosing taxa with minimum distance would be wrong.

	x_1	x_2	x_3	x_4
x_1	0	3	5	6
x_2		0	6	5
x_3			0	9
x_4				0



Neighbor-Joining

For the Neighbor-Joining method one defines

$$N_{i,j} = d(i,j) - (r_i + r_j)$$

where $r_i = \frac{1}{n-2} \sum_{x \in L} d(i,x)$, L is the current set of nodes in the tree, and $n = |L|$. r_i approximates the average distance to all other leaves: instead of dividing by n, one divides by $n-2$.

Theorem 6

If d is additive w.r.t. some tree T, then the two leaves i and j for which $N_{i,j}$ is minimal are neighbors in T. □

- This result ensures that the neighbor-joining algorithm will correctly reconstruct a tree from its additive distances.
- Unlike for the UPGMA-method, ultrametric properties (or something similar), are not required for the distance function.

Example 7

Let us illustrate this result using the following distance function. We have $L = \{x_1, x_2, x_3, x_4\}$ and so $n = 4$ and thus:

$$r_{x_i} = \frac{1}{n-2} \sum_{x \in L} d(x_i, x) = \frac{1}{2} \sum_{j=1}^{4} d(x_i, x_j)$$

	x_1	x_2	x_3	x_4
x_1	0	3	5	6
x_2		0	6	5
x_3			0	9
x_4				0

$$r_{x_1} = \frac{1}{2}(3 + 5 + 6) = 7$$

$$r_{x_2} = \frac{1}{2}(3 + 6 + 5) = 7$$

$$r_{x_3} = \frac{1}{2}(5 + 6 + 9) = 10$$

$$r_{x_4} = \frac{1}{2}(6 + 5 + 9) = 10$$

Example 7

$$\begin{aligned}
\text{and } N_{x_1,x_2} &= d(x_1,x_2) - (r_{x_1} + r_{x_2}) = 3 - (7+7) = -11 \\
N_{x_1,x_3} &= d(x_1,x_3) - (r_{x_1} + r_{x_3}) = 5 - (7+10) = -12 \\
N_{x_1,x_4} &= d(x_1,x_4) - (r_{x_1} + r_{x_4}) = 6 - (7+10) = -11 \\
N_{x_2,x_3} &= d(x_2,x_3) - (r_{x_2} + r_{x_3}) = 6 - (7+10) = -11 \\
N_{x_2,x_4} &= d(x_2,x_4) - (r_{x_2} + r_{x_4}) = 5 - (7+10) = -12 \\
N_{x_3,x_4} &= d(x_3,x_4) - (r_{x_3} + r_{x_4}) = 9 - (10+10) = -11
\end{aligned}$$

$$\text{and so } N = \begin{array}{c|cccc} & x_1 & x_2 & x_3 & x_4 \\ x_1 & 0 & -11 & -12 & -11 \\ x_2 & & 0 & -11 & -12 \\ x_3 & & & 0 & -11 \\ x_4 & & & & 0 \end{array}$$

The matrix N attains a minimum value for the pairs (x_1, x_3) and (x_2, x_4) and the corresponding leaves are indeed neighbors, as required.

Algorithm 2 (Neighbor-Joining)

Input: Distance matrix d; **Output:** Phylogenetic tree T
Initialization: Let T consist of leaf nodes, one for each taxon. $L \leftarrow T$.
Iteration: Compute N from the current distance function. Pick a pair $i, j \in L$, s.t. $N_{i,j}$ is minimal. Define a new node k and set

$$d(k, m) = \frac{1}{2}(d(i, m) + d(j, m) - d(i, j)) \tag{2}$$

for all $m \in L \setminus \{i, j\}$. Add k to T and construct edges from k to i and to j. The edge length are:

$$\begin{aligned} d(i, k) &= \frac{1}{2}(d(i, j) + r_i - r_j) \\ d(j, k) &= d(i, j) - d(i, k) \end{aligned} \tag{3}$$

where $r_i = \frac{1}{n-2} \sum_{x \in L} d(i, x)$. Remove i and j from L and add k to L.
Termination: When L consists of only two elements i and j, add the remaining edge between i and j, with length $d(i, j)$.

Example 8

Let d_0 be given, as shown below. The following shows the different distance functions computed during the neighbor-joining algorithm, and the corresponding N-matrix computed from these.

$$d_0 = \begin{array}{c|cccc|c} & x_1 & x_2 & x_3 & x_4 & r_i \\ x_1 & 0 & 3 & 5 & 6 & 7 \\ x_2 & & 0 & 6 & 5 & 7 \\ x_3 & & & 0 & 9 & 10 \\ x_4 & & & & 0 & 10 \end{array} \Rightarrow N^0 = \begin{array}{c|cccc} & x_1 & x_2 & x_3 & x_4 \\ x_1 & & -11 & \underline{-12} & -11 \\ x_2 & & & -11 & -12 \\ x_3 & & & & -11 \\ x_4 & & & & \end{array}$$

$$d_1 = \begin{array}{c|ccc|c} & x_1, x_3 & x_2 & x_4 & r_i \\ \{x_1, x_3\} & 0 & 2 & 5 & 7 \\ x_2 & & 0 & 5 & 7 \\ x_4 & & & 0 & 10 \end{array} \Rightarrow N^1 = \begin{array}{c|ccc} & \{x_1, x_3\} & x_2 & x_4 \\ \{x_1, x_3\} & & -12 & -12 \\ x_2 & & & \underline{-12} \\ x_4 & & & \end{array}$$

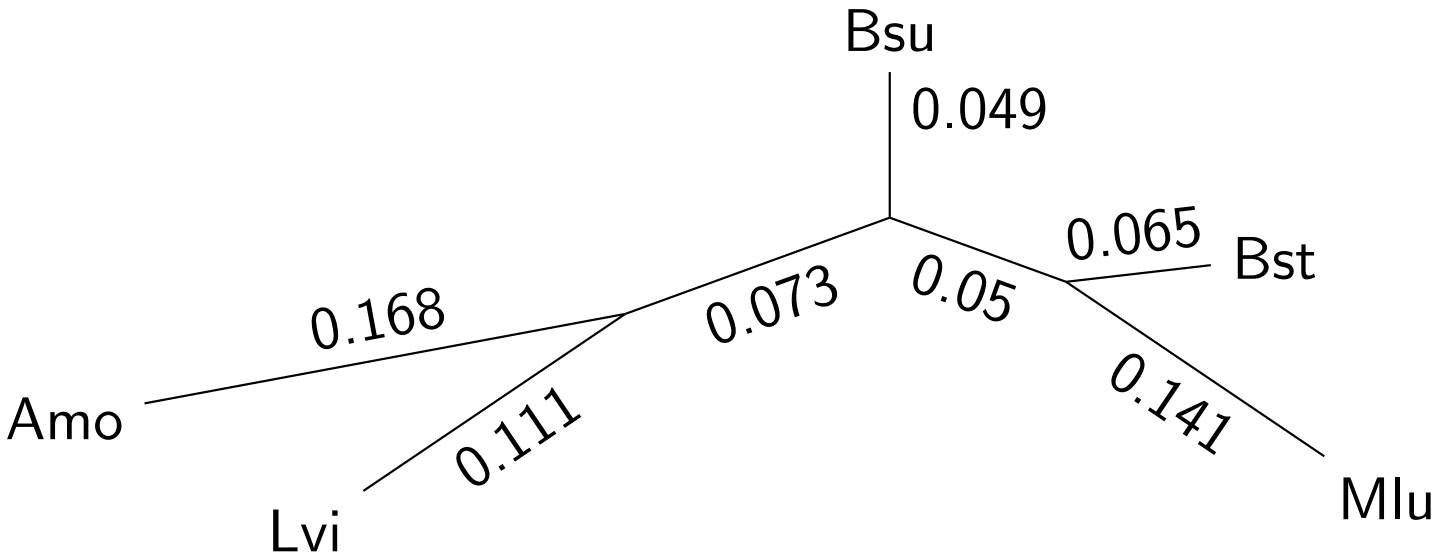
$$d_2 = \begin{array}{c|cc|c} & \{x_1, x_3\} & \{x_2, x_4\} & r_i \\ \{x_1, x_3\} & 0 & 1 & 0 \\ \{x_2, x_4\} & & 0 & 0 \end{array}$$

Example 9

For the following distance function (which was already used in Example 5)

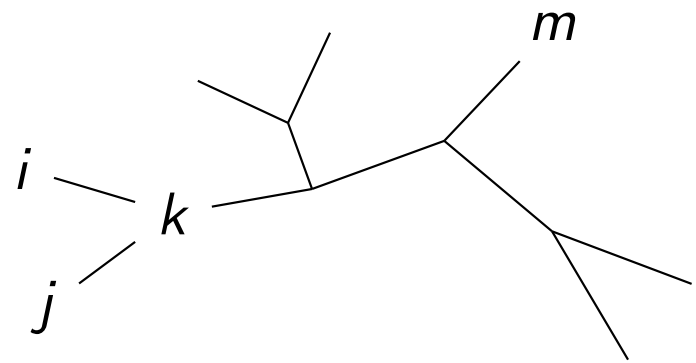
	Bsu	*Bst*	*Lvi*	*Amo*	*Mlu*
Bsu	0.0	0.1715	0.2147	0.3091	0.2326
Bst		0.0	0.2991	0.3399	0.2058
Lvi			0.0	0.2795	0.3943
Amo				0.0	0.4289
Mlu					0.0

the neighbor-joining algorithm delivers the following phylogenetic tree (with some edge length rounded):



Neighbor-Joining

Recall that k is the new parent for i and j and we need to compute the distance from k to any other leaf $m \notin \{i, j\}$, see figure right.



Given $d(i, m)$, $d(j, m)$, and $d(i, j)$, and due to the assumption that d is additive, we obtain

$$\begin{aligned} d(i, m) &= d(i, k) + d(k, m) \\ d(j, m) &= d(j, k) + d(k, m) \\ d(i, j) &= d(i, k) + d(j, k) \end{aligned}$$

$$\begin{aligned} \text{which implies } d(k, m) &= 0.5 \cdot 2 \cdot d(k, m) \\ &= 0.5 \cdot (d(i, m) - d(i, k) + d(j, m) - d(j, k)) \\ &= 0.5 \cdot (d(i, m) + d(j, m) - (d(i, k) + d(k, j)) \\ &= 0.5 \cdot (d(i, m) + d(j, m) - d(i, j)) \end{aligned}$$

This is how $d(k, m)$ is calculated in Equation (2) of Algorithm 2.

Neighbor-Joining

Let us now consider the update formula $d(i,k) = \frac{1}{2}(d(i,j) + r_i - r_j)$ in Algorithm 2. By definition,

$$\begin{aligned} r_i &= \frac{1}{n-2} \sum_{x \in L} d(i,x) \\ &= \frac{1}{n-2} \sum_{x \in L \setminus \{i,j\}} d(i,x) + \frac{1}{n-2}(d(i,j) + d(i,i)) \\ &= \underbrace{\frac{1}{n-2} \sum_{x \in L \setminus \{i,j\}} d(i,x)}_{q_i} + \frac{1}{n-2} d(i,j) \end{aligned}$$

In other words, r_i is the average distance q_i to all nodes (except for i and j) plus $\frac{1}{n-2} d(i,j)$.

Neighbor-Joining

$$\begin{aligned}
\text{Now } r_i - r_j &= \frac{1}{n-2}\sum_{x\in L\setminus\{i,j\}} d(i,x) + \frac{1}{n-2}d(i,j) - \left(\frac{1}{n-2}\sum_{x\in L\setminus\{j,i\}} d(j,x) + \frac{1}{n-2}d(j,i)\right)\\
&= \frac{1}{n-2}\sum_{x\in L\setminus\{i,j\}} d(i,x) - \frac{1}{n-2}\sum_{x\in L\setminus\{j,i\}} d(j,x) + \frac{1}{n-2}d(i,j) - \frac{1}{n-2}d(i,j)\\
&= \frac{1}{n-2}\sum_{x\in L\setminus\{i,j\}} d(i,x) - \frac{1}{n-2}\sum_{x\in L\setminus\{j,i\}} d(j,x)\\
&= q_i - q_j
\end{aligned}$$

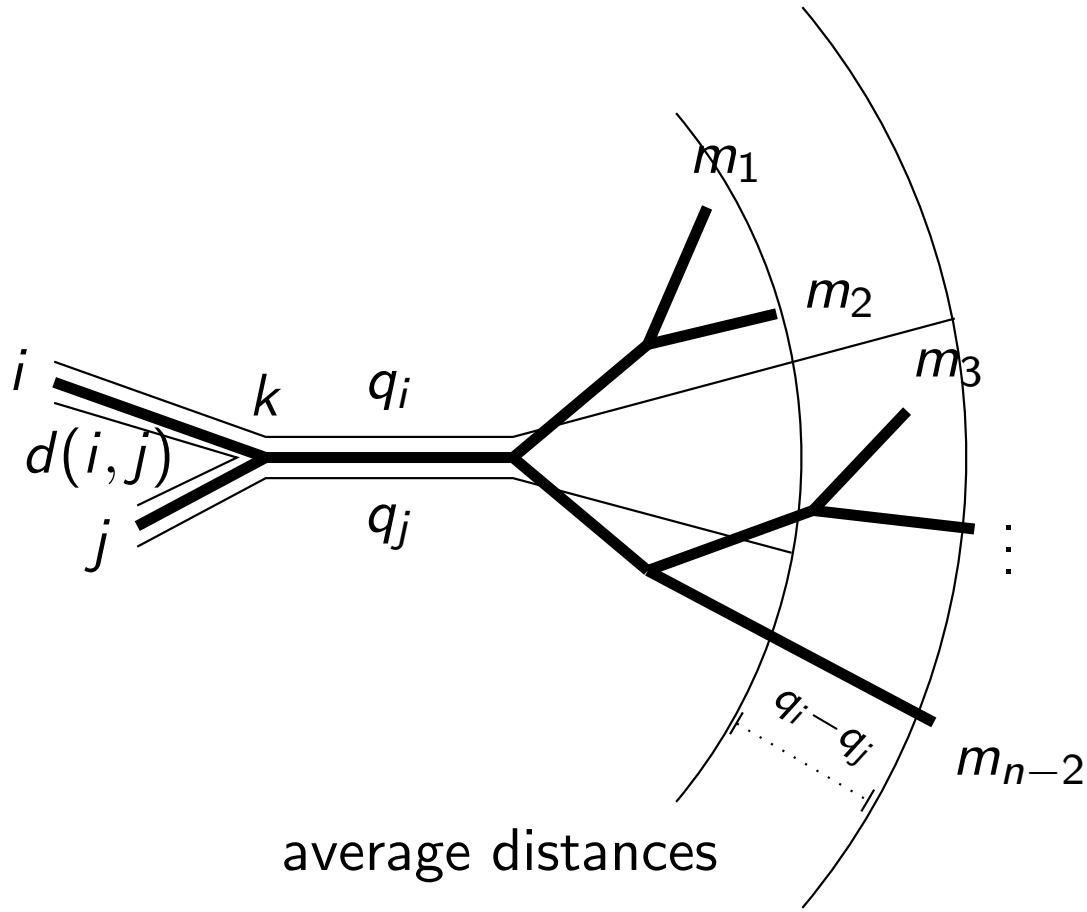
Hence we obtain

$$d(i,k) = \frac{1}{2}(d(i,j) + r_i - r_j) = \frac{1}{2}(d(i,j) + q_i - q_j)$$

That is, $d(i,k)$ is half of the distance from i to j plus the difference of the average distances to i and to j, see Figure 7 for a graphical explanation.

Figure 7: Determining $d(i,k) = \frac{1}{2}(d(i,j) + q_i - q_j)$.

The thick edges represent the tree consisting of the n nodes $\{i, j, m_1, \ldots, m_{n-2}\}$. The thin straight lines represent the distances involved in the computation of $d(i,k)$. As j is closer to k, the average distance q_j is smaller than q_i and so the thin straight line labeled by q_j ends on the inner circle, while the thin straight line labeled by q_i ends on the outer circle. The difference between q_i and q_j is shown as a dotted line between the circles.



Application of Neighbor-Joining

- Given an additive distance function $d : X \times X \to \mathbb{R}^0_+$, neighbor-joining is guaranteed to reconstruct a phylogenetic tree T_X, such that d is additive w.r.t. T_X.
- Unfortunately, in practice we are usually not given a distance function that is additive, but rather the distance function is usually obtained very indirectly by comparing sequence data generated along the tree.
- Such data is rarely additive.
- Nevertheless, the neighboring-joining method is often applied to such data and has proven to be a fast, useful and robust tree reconstruction method.