Phylogeny Tree Reconstruction

1 Introduction

The topic of this class is the mathematical modeling of evolution. Central questions are:

- After aligning a set of sequences, what is a rigorous way to estimate evolutionary distances?
- How can we reconstruct a phylogeny tree?

Evolution introduces changes from each copy of the DNA to the next. These changes are usually very small events: Deletions or mutations of a few letters. However, they also come in very large or complex forms as tandem or distance duplications. Many of these large mutations are repeats. The mechanisms behind many of these large events are yet unexplored. These larger events occur often, but they are comparatively rare compared to smaller insertions/deletions/mutations. In order to figure out evolutionary distance today, it is therefore most useful to observe point mutations.

2 Nomenclature

One of the main ways proteins evolve is through gene duplication. The vast majority of human genes arose through gene duplication. Even whole genomes can duplicate. After duplication of a protein-coding gene, there are two pieces of DNA that produce the same protein. Therefore, evolution allows one of the copies to change its function while the other one retains its original function. Even small changes in protein structure can bring about great differences in protein functionality.

Through the same method (duplication), gene regulatory elements can also undergo evolution. When a regulatory region is copied and evolves, the conditions under which the associated protein coding genes are produced can evolve. Most of gene regulatory elements are encoded in promoters and enhancers upstream of the protein coding sequence. When these are duplicated, and one of the clones is modified, then expression regulation can evolve as well.

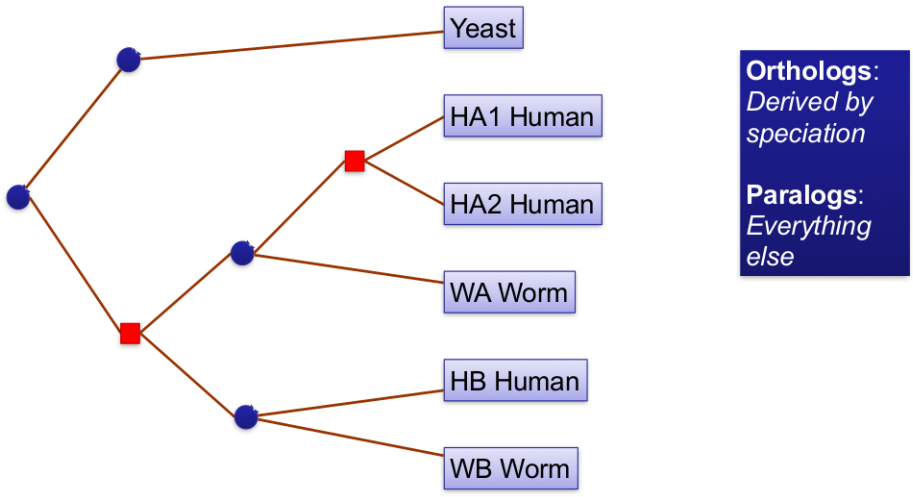
The evolutionary trees seen so far depict the lineage from ancestral species to the species today. But these trees can be extended to include information about individuals in the same species: For example, a genetic regions might be duplicated from father to son. We want to distinguish between these two types of branching. In our phylogenetic trees, squares signify duplication, and circles signify protein speciation.

Every pair of species derived from the same ancestor are called **homologs**. These can be split into **orthologs** and **paralogs**:

- **Orthologs** are loci derived from speciation. Definition: Two leaf nodes in the phylogenetic tree are orthologous if the nearest common ancestor node in the phylogenetic tree is a speciation node.

- **Paralogs** are all other loci.

Examples:



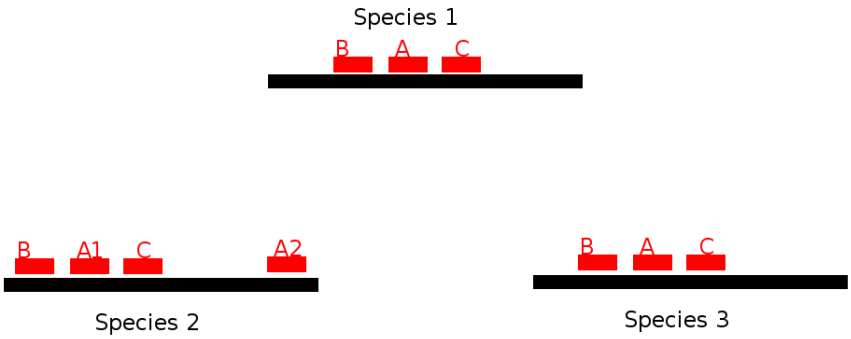
HB human and WB worm are orthologs. WB and WA are paralogs. WB and HA1,2 are paralogs. All human genes are orthologs to yeast.

3 Orthology, Paralogy, Imparalogs, Outparalogs

The triangular map shows even more ways two species could be related.

Xenology (Greek ξένος=foreigner): Sequences being related through something foreign. A common example of this are pieces of DNA that were originally proliferated by some virus. A retrovirus procreates by integrating its genetic material into the host's DNA. If the virus finds a way into some species' germline, then two seemingly unrelated species can become xenologous. Somatic cells can be xenologous as well (e.g., after becoming infected with HIV).

Orthology: As discussed above, orthology means that two sequences are similar after a speciation event. There are two subclasses of orthology: **Topoorthology** and **Monotopoorthology**. (Greek τόπος =place/location).



In the example depicted, there was a speciation event from species 1 into species 2 and 3. In species 2, the gene A was copied into two distinct loci, leading to versions A1 and A2 of the gene. In species 3, only one version of A continues to exist. In species 2, both A1 and A2 are common with A, so both of them are orthologous. In species 2, A1 is also in place (next to B and C), so it is topoorthologous. In species 3, A was not copied and it is in place, and therefore it is monotopoorthologous.

Paralogy: As discussed above, paralogs derive from duplication events. Paralogy can be divided into **imparalogy** and **outparalogy**. A detailed discussion of these terms would be out of scope here. Imparalogy and outparalogy derive from the question if the duplication events happened within the evolutionary scope under scrutiny. When studying eucaryotes, then all human genes are imparalogs. When studying only certain humans, then some human genes are outparalogs.

4 Phylogeny Trees

The ENCODE project, currently in its third phase, deals with annotating elements in the human genome. Examples of possible annotations include:

- What genes are expressed in which cell lines under what conditions
- Which enhancers and promoters are involved
- Which transcription factors are involved in gene expression
- Finding all proteins that bind to DNA
- Exploring DNA methylation (which controls gene activity)

The first phase of the ENCODE project set out to analyze 1% of the human genome. Among this 1% were some random regions and some regions known to be very important. This 1% was sequenced in many species. ENCODE then connected this 1% of the genome and its counterpart in all species.

Looking at the phylogenetic tree (Figure 1), you can easily find the closest common ancestor for each pair of species. The edge lengths are also well measured, and they represent evolutionary distance. evolutionary speeds of the different species. Observe that the human edge is barely

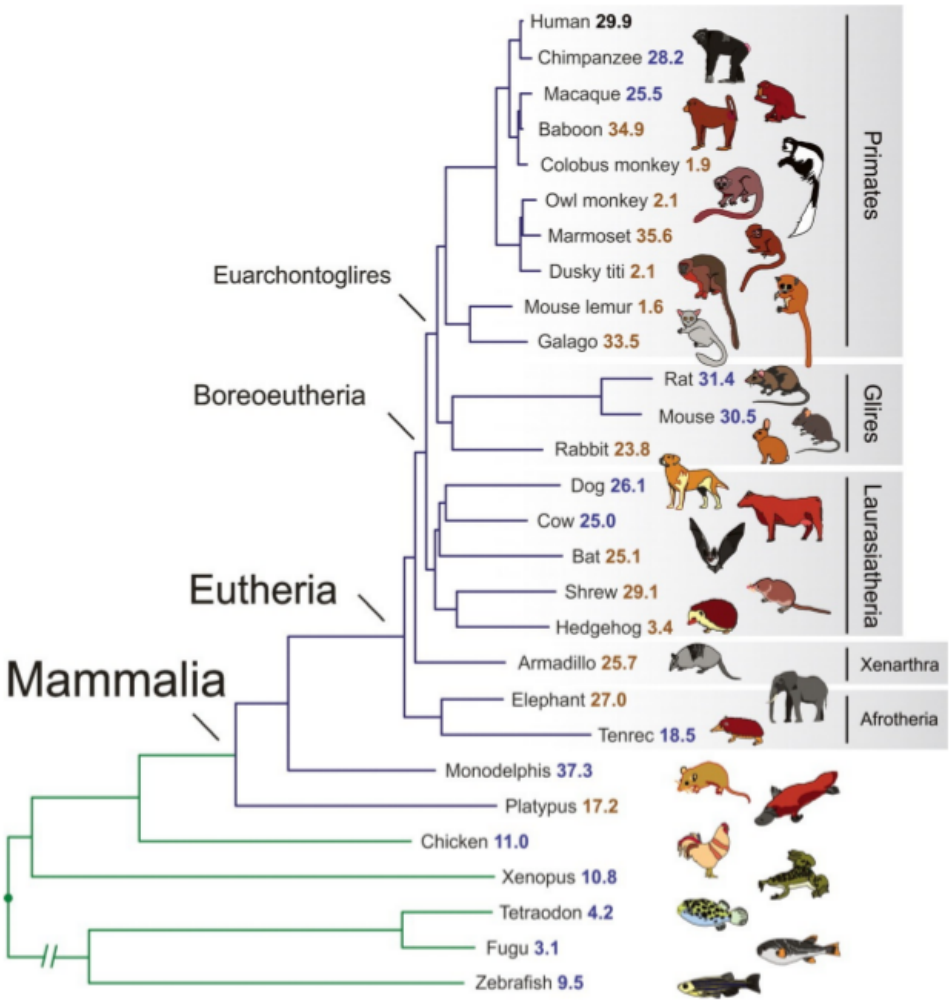


Figure 1: Phylogeny tree of species sequenced so far.

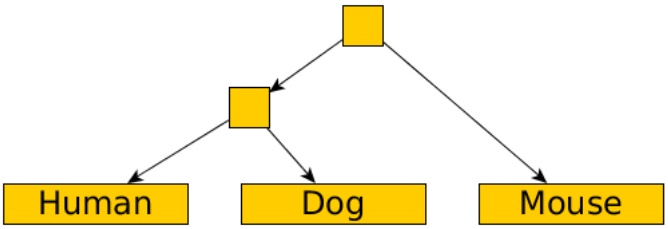


Figure 2: Phylogenetic tree suggested by sequence alignment (wrong).

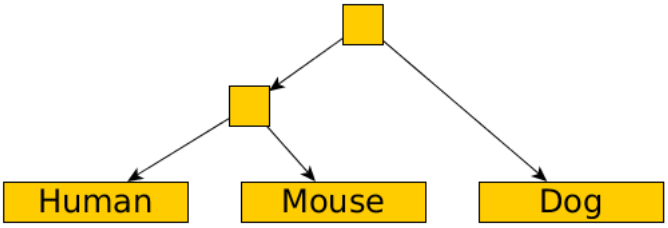


Figure 3: Correct phylogenetic tree after construction of an evolutionary model.

visible. The rodent edges are much longer, in comparison. What is the measure of evolutionary distance?

We set out to calculate what the evolutionary distance between a pair of sequences is. Initially, there was much controversy over whether the mouse is more related to the human or the dog. By current genomic data, the hypothesis that the mouse is more closely related to the human is better supported. The reasons for this controversy was that, looking only at sequence alignment, human and dog seem closer than human and mouse.

However, the correct placement is such that the dog is an outgroup compared to human and mouse.

The two questions addressed here are:

- How can we build an evolutionary tree from sequence data?
- How long should the edges in this tree be?

After building the tree, determining the edge lengths is easy: Each edge is proportional to the substitutions per site. The average number of substitutions per site is called the substitution rate, and it is proportional to the edge lengths in the tree.

We want to create phylogeny trees through sequence alignment. In order to build a phylogeny tree, we first take the alignment and translate it to substitutions per site.

Consider what happens at a given position at a given time frame Δt. Imagine we start at a given letter in the genome, i.e. at one specific site. The vector $P(t)$ gives the probabilities for the letter being either A, C, T or G at a time t. We write p_A to denote the element A of vector $P(t)$. (Similarly, p_x is defined for all other bases $x \in \{C, G, T\}$.) Time will be modeled as a continuous variable here—there are so many generations involved in evolution that time is approximately continuous. The rate of a given letter changing is given by $\lim_{\Delta t \to 0} P(t + \Delta t)$. The variable μ_{AC}

- Jukes-Cantor

$$Q = \begin{pmatrix} * & \frac{\mu}{4} & \frac{\mu}{4} & \frac{\mu}{4} \\ \frac{\mu}{4} & * & \frac{\mu}{4} & \frac{\mu}{4} \\ \frac{\mu}{4} & \frac{\mu}{4} & * & \frac{\mu}{4} \\ \frac{\mu}{4} & \frac{\mu}{4} & \frac{\mu}{4} & * \end{pmatrix}$$

- Kimura

$$Q = \begin{pmatrix} * & \kappa & 1 & 1 \\ \kappa & * & 1 & 1 \\ 1 & 1 & * & \kappa \\ 1 & 1 & \kappa & * \end{pmatrix}$$

- Felsenstein

$$Q = \begin{pmatrix} * & \pi_T & \pi_T & \pi_T \\ \pi_C & * & \pi_C & \pi_C \\ \pi_A & \pi_A & * & \pi_A \\ \pi_G & \pi_G & \pi_G & * \end{pmatrix}$$

- HKY

$$Q = \begin{pmatrix} * & \kappa\pi_T & \pi_T & \pi_T \\ \kappa\pi_C & * & \pi_C & \pi_C \\ \pi_A & \pi_A & * & \kappa\pi_A \\ \pi_G & \pi_G & \kappa\pi_G & * \end{pmatrix}$$

Figure 4: Q matrices of different evolutionary models.

denotes the rate of transition from A to C per unit time. μ_A denotes the rate that A changes to a different letter. (μ is defined similarly for all other bases and pairs of bases.) We have

$$\mu_A = \mu_{AC} + \mu_{AG} + \mu_{AT}.$$

Additionally,

$$p_A(t + \Delta t) = p_A(t) - p_A(t)\mu_A\Delta t + p_C(t)\mu_{CA}\Delta t + p_G(t)\mu_{GA}\Delta t + p_T(t)\mu_{TA}\Delta t.$$

In matrix notation, this reads:

$$P(t + \Delta t) = P(t) + QP(t)\Delta t,$$

where Q is the substitution rate matrix:

$$\begin{pmatrix} -\mu_A & \mu_{GA} & \mu_{CA} & \mu_{TA} \\ -\mu_{AG} & -\mu_G & \mu_{CG} & \mu_{TG} \\ -\mu_{AC} & \mu_{GC} & -\mu_C & \mu_{TC} \\ -\mu_{AT} & \mu_{GT} & \mu_{CT} & -\mu_T \end{pmatrix}$$

As Δt goes to 0, we arrive at a differential equation:

$$\frac{\partial}{\partial t}P(t) = QP(t),$$

which we solve for $P(t)$. Here, Q is the probability distribution over A, C, T and G at each position. The matrix Q represents an evolutionary model. Some evolutionary models work better than others. In the simplest model, all letters are equally likely, and all transitions from letter to letter are equally likely. The corresponding Q matrix is called the *Jukes-Cantor* model.

In the *Kimura* model, mutations from purines (A and G) to pyrimidines (C and T) and vice versa are penalized. This corresponds to a biological reality: Mutations from A to G (and vice versa) and from C to T (and vice versa) are more likely than all other mutations. Mutations from purine to purine and from pyrimidine to pyrimidine are called transitions. All other mutations are called transversions, and they have a lower probability in this model.

In the *Felsenstein* model, different stationary probabilities are assigned to each letter. The *HKY* model is a combination of the Kimura and the Felsenstein models.

Now equipped with evolutionary models, we can solve the differential equation. Because the Jukes-Cantor model is the simplest of the presented models, let's solve the differential equation with the Jukes-Cantor model. The function $r(t)$ signifies the rate of staying at the same letter at time t. The function $s(t)$ signifies the mutation rate. So let

$$r(t) = P_{AA}(t) = P_{CC}(t) = P_{GG}(t) = P_{TT}(t)$$

and

$$s(t) = P_{AC}(t) = \cdots = P_{TG}(t).$$

Then

$$\frac{\partial}{\partial t} r(t) = -\frac{3}{4} r(t)\mu + \frac{3}{4} s(t)\mu$$

and

$$\frac{\partial}{\partial t} s(t) = -\frac{1}{4} s(t)\mu + \frac{1}{4} r(t)\mu.$$

Solving these differential equations yield

$$r(t) = \frac{1}{4}(1 + 3e^{-\mu t})$$

and

$$s(t) = \frac{1}{4}(1 - e^{-\mu t}).$$

Now we can plug in some values for t. At $t = 0$, the value of r is 1. As time goes towards infinity, the term becomes $\frac{1}{4}$: any letter can change to any letter with equal probability.

However, we are actually not that interested in calculating the effects of the model after a given time. Rather, we want to start with given effects and compute the time it took for the respective changes to happen: Assume we are given two sequences and an alignment. From the alignment, we can infer some information about the similarity, but we do not know the evolutionary distance between the two sequences. To do this, we want to solve the above equations (here, from the Jukes-Cantor model) for t:

$$\begin{aligned} r(t) = 1 - p &= \frac{1}{4}(1 + 3e^{-\mu t}) \\ p &= \frac{3}{4} - \frac{3}{4} e^{-\mu t} \\ \frac{3}{4} - p &= \frac{3}{4} e^{-\mu t} \\ 1 - \frac{4p}{3} &= e^{-\mu t} \\ \mu t &= -ln(1 - \frac{4p}{3}) \end{aligned} \tag{1}$$

Letting $d = \frac{3}{4}\mu t$, denoting substitutions per site,

$$d = -\frac{3}{4} ln(1 - \frac{4}{3}p).$$

Thus, given an alignment, we can calculate the substitutions per site if evolution followed the Jukes-Cantor model (or any other model more realistic than Jukes-Cantor).

4.1 Determining the Initial Probability

Calculating the initial probability vector $P(t)$ is difficult, as alignment errors can confound us during the calculation. Additional questions that may arise are: What does it mean for two letters to be different? Do deletions count in this calculation? A lot of effort has gone into calculating the initial probability $P(T)$ in the best way. As an example for finding the neutral rate of evolution, a good way might be to count the mutations in fourfold degenerate codons. These are codons, i.e., triplets of amino acids, whose third letter is irrelevant in determining the translated amino acid: All versions of the codon code for the same amino acid regardless of the value of the third letter. It is assumed that the mutation rate in third letters of four-fold degenerate codons is neutral.

However, there are a few caveats to this: There could be regulatory or structural elements that bind DNA and that have some bias w/r/t function or efficiency to one of these forms. But still, four-fold degenerate sites are considered neutral. Then the ratio of letters at four-fold degenerate sites can be taken as the initial probability $P(t)$.

A different way of finding the initial probability $P(t)$ is through repeat analysis. After sequencing a number of pairs of humans and mice, align all genes. Then find a repeat element that, presumably, does not have a function, and that has a monotopoorthologous repeat copy across the two species. Then the evolution of this nonfunctional repeat element is assumed to be close to the neutral rate. Other ways to estimate the initial probability $P(t)$ also depend on discovering regions in different species that are believed to evolve close to the neutral rate.

4.2 Building the Phylogeny Tree Based on Evolutionary Distances

Assume now we have multiple sequences $x_1, \ldots, x_n$. To build the phylogeny, apply the following steps in correct order:

1. Pick your favorite evolutionary model. (Jukes-Cantor is good enough for practical purposes.)
2. Align x_i and x_j for all pairs i and j.
3. Estimate the evolutionary distance between each of these pairs.
4. Using these distances, build the phylogeny tree.

We can build the phylogeny tree from the matrix of evolutionary distances between all pairs of sequences x_i, x_j. The first method discussed is called **UPGMA**.

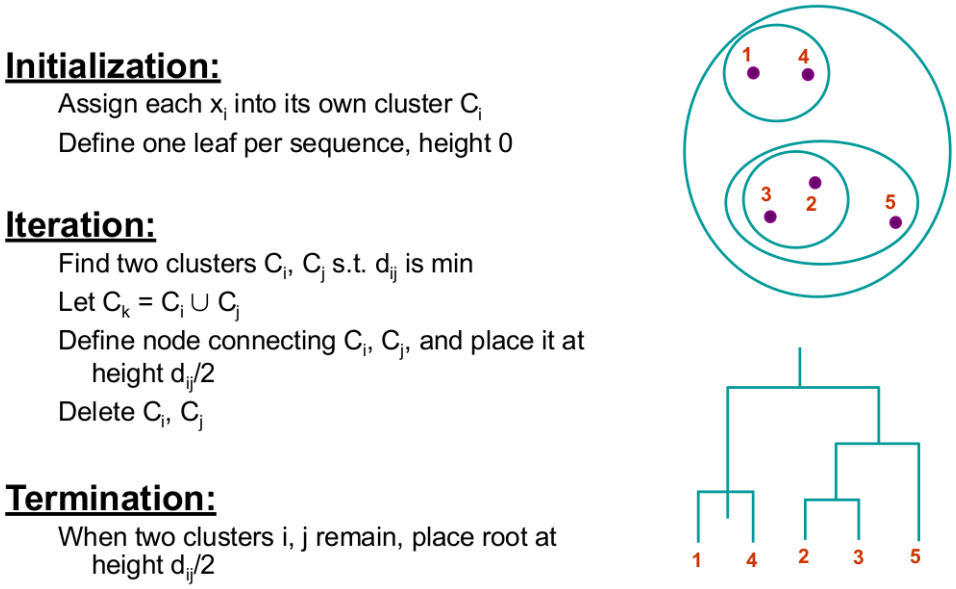


Figure 5: Sketch of the incremental tree construction in UPGMA.

4.3 UPGMA

UPGMA stands for "unweighted pair group method using arithmetic averages". It is also known as the **Average Linkage Method**. This method assumes that evolutionary distances are equivalent to the chronological distances. According to this, all phylogenetic trees would satisfy the property that the length of each leaf node to the root node would be precisely the same.

Given two disjoint clusters C_i, C_j of sequences,

$$d_{ij} = \frac{1}{|C_i| \times |C_j|} \sum_{\{p \in C_i, q \in C_j\}} d_{pq} .$$

It is easy to compute the distances between clusters incrementally using the following formula:

$$d_{kl} = \frac{d_{il}|C_i| + d_{ji}|C_j|}{|C_i| + |C_j|}.$$

Using this incremental calculation of distances, the phylogenetic tree can be constructed quite fast. A sketch of the algorithm is given in Figure 5. Thus, at any given point, we only need to recompute the distance of the new cluster to the rest.

4.3.1 Average Linkage Example

In figure 6, an example of incremental cluster construction is presented. The algorithm starts with each sequence in its own cluster. Then it finds the two clusters for which the distance is minimal, which is yz here, and merges them by applying the two-cluster-distance rule above. In the graph, node 1 is constructed as the ancestor of y and z and clusters y and z are deleted from the matrix and replaced by yz. Similarly, the algorithm continues to merge x into yz, creating node 2 and cluster xyz. ...

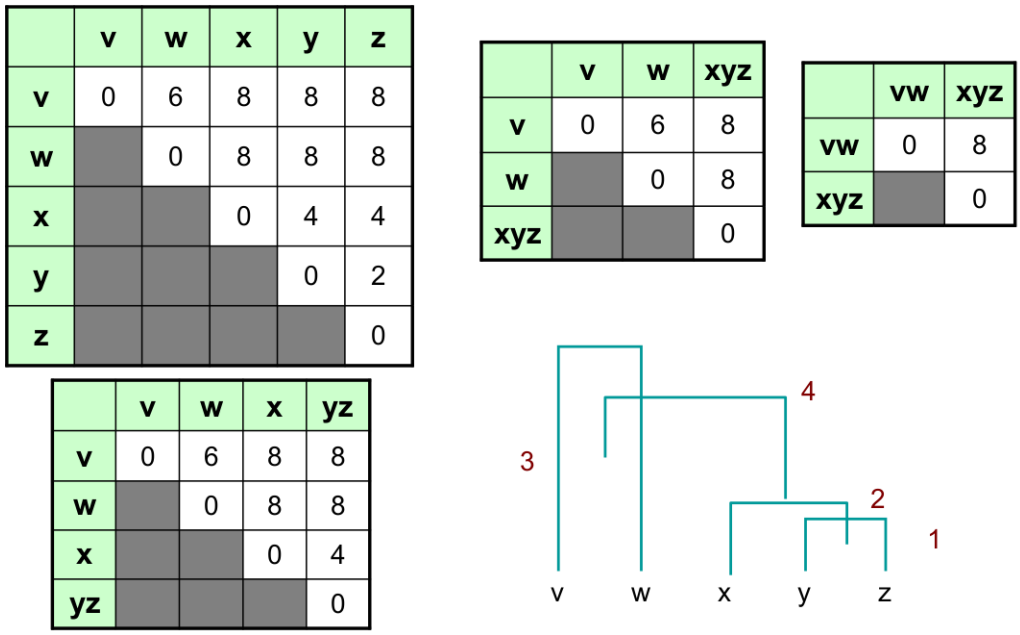


Figure 6: Example of incremental tree construction in UPGMA.

4.3.2 Ultrametric Distances and Limitations of UPGMA

A distance function $d(\cdot, \cdot)$ is ultrametric if for any three distances $d_{ij} \leq d_{ik} \leq d_{jk}$ it is true that

$$d_{ij} \leq d_{ik} = d_{jk}.$$

Average Linkage is guaranteed to reconstruct a binary tree correctly if the distance function is ultrametric. (Why? Informal argument: Looking at a subtree consisting of any three species, where y and z are closer and x is the outgroup, the distances yx and zx will be the same in a constant-rate tree as constructed by UPGMA, and they will be greater than the distance yz.) This assumes that the molecular clock has a constant rate across all species.

However, UPGMA will produce highly incorrect results if applied to trees that are actually very unbalanced, such as seen in Figure 7.

4.4 Additive Distances for Reconstruction

A distance measure is additive if the distance between any pair of leaves is the sum of lengths of edges connecting them. Given a tree T and an additive distance matrix d_{ij}, we can uniquely reconstruct all edge lengths: Notice that for any two leaves i and j with a common parent k, we can use a fourth leaf m to properly distance i from j and k: The parent node k is placed at distance

$$d_{km} = \frac{1}{2}(d_{im} + d_{jm} - d_{ij}) \tag{2}$$

This way, all distances can be reconstructed given a distance matrix and a tree without edge weights.

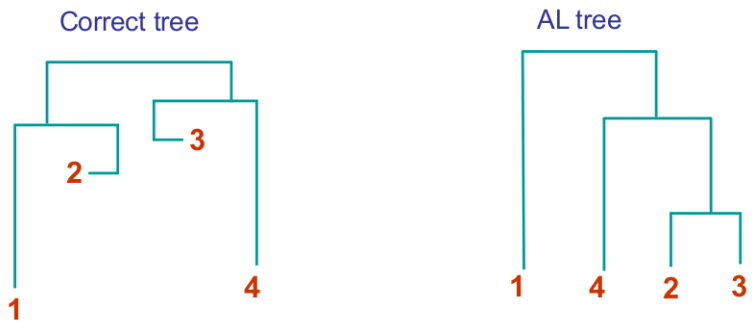


Figure 7: Example where UPGMA reconstructs a wrong tree because of its incorrect assumption that the evolutionary rate is the same across all species and linearly related to time.

However, in most cases, we do not actually have the tree and just need to figure out edge lengths. Usually, we just have a distance matrix and must use this to construct both the tree and infer edge lengths.

Figure 8 shows an example of edge length reconstruction given a tree and a distance matrix.

4.5 Neighbor Joining

Neighbor joining provably reconstructs the correct tree when given an additive distance matrix. It even constructs good trees if the distance is not additive or if there are errors in the measurements. It relies on the neighbor joining formula:

$$D_{ij} = (N-2)d_{ij} - \sum_{k \neq i} d_{ik} - \sum_{k \neq j} d_{jk},$$

where N is the number of species.

In the original matrix of distances denoted by d_{ij}, it is not possible to assign the i and j with minimal distance as neighbors because they are not necessarily neighboring leaves with a common parent. In the above example, Nodes 1 and 3 are closest, but they are not neighboring leaves. We need to construct a new distance matrix where the nodes with minimal distance are actually neighboring leaf nodes. This distance matrix is given by the neighbor joining formula.

Figure 9 shows how the neighbor joining formula succeeds in creating a distance matrix in which i and j are neighbors if D_{ij} are minimal.

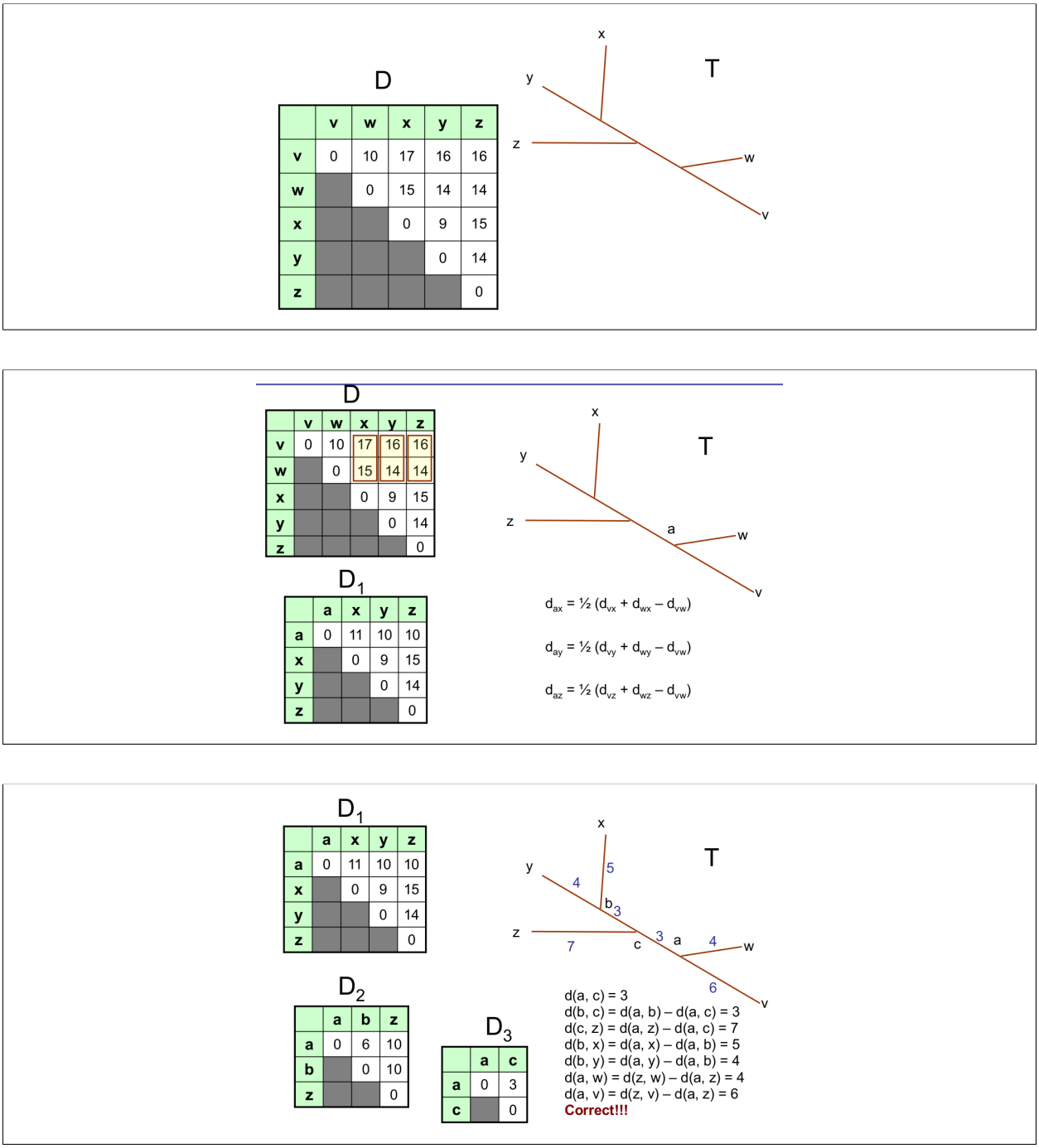
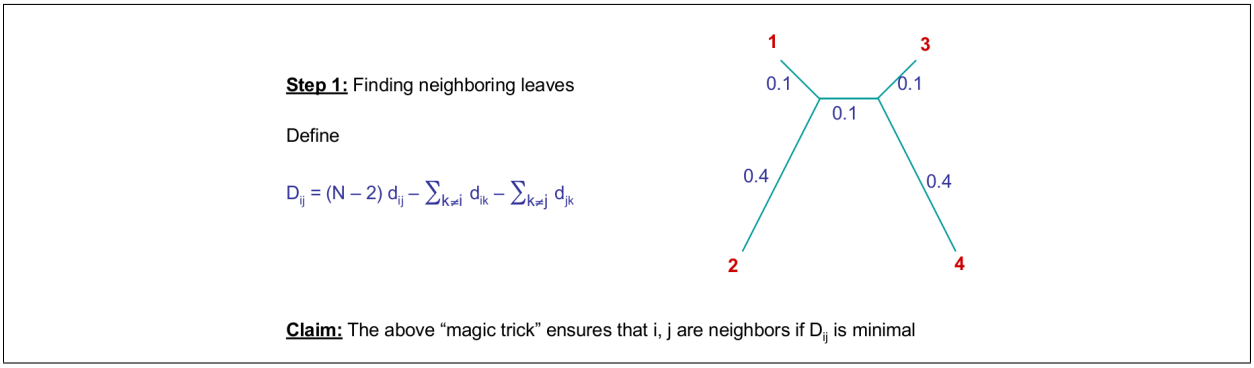
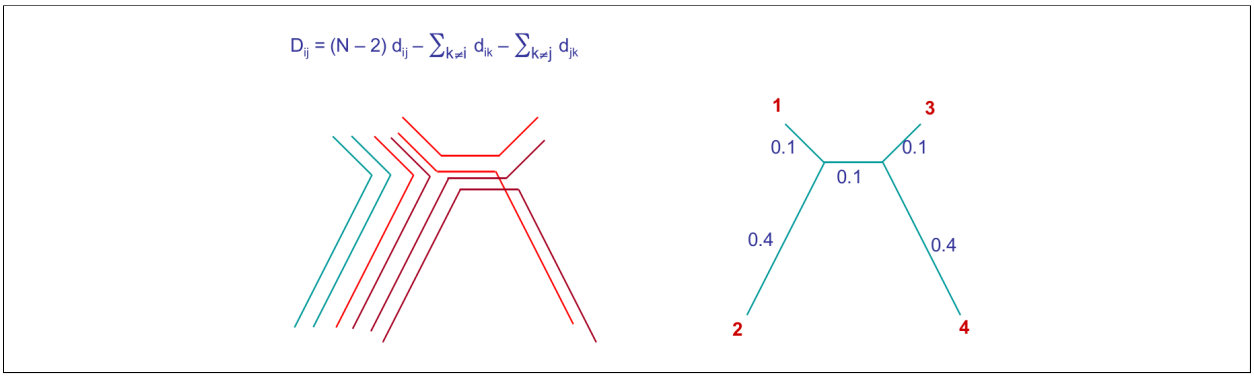


Figure 8: Reconstruction of a edge lengths given a tree T and an additive distance matrix D. **Above**: We have T and D, but not the edge lengths. **Middle**: Use equation 2 to determine edge lengths one by one using triples nodes (like a, w and v in this example) plus an outgroup (like x, y and z in this example). **Below**: Applying this formula for all edges in the tree magically yields the correct edge lengths, as can be verified by summing over the edge weights along any path and comparing to the original distance matrix D.





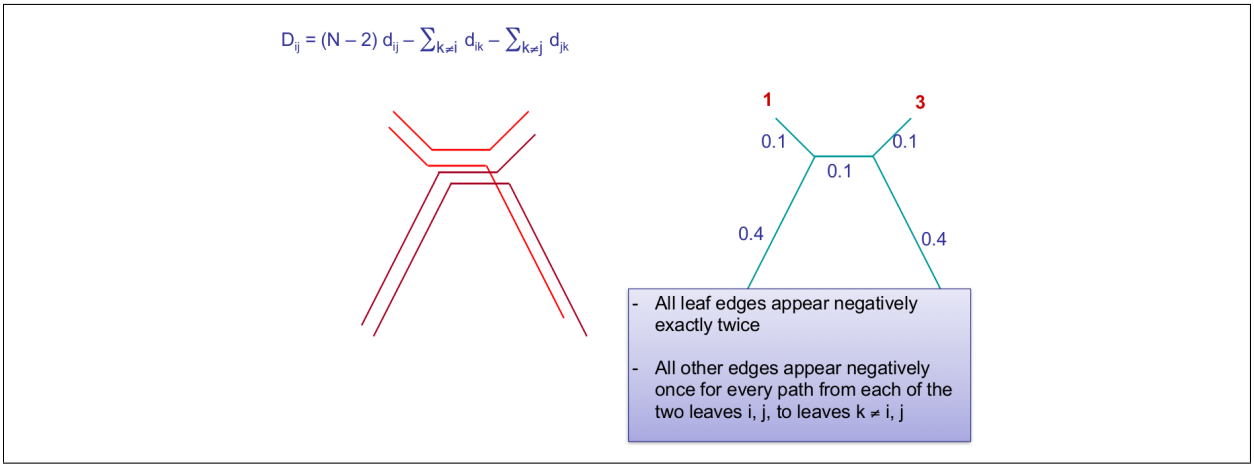


Figure 9: **Above**: Tree with edge lengths, but two nodes are not necessarily neighbors if their distances are minimal: E.g., 1 and 2 are neighbors, but 1 and 3 are actually closer in distance. **Middle**: Neighbor joining adds edges for every two actual neighbors and subtracts edges for all pairs of nodes i, k and j_k. **Below**: This ensures that all leaf edges appear negatively exactly twice, and all other edges appear negatively exactly once for every path from each of the two leaves i, j to all other leaves, thus ensuring that neighboring nodes have a minimal distance.