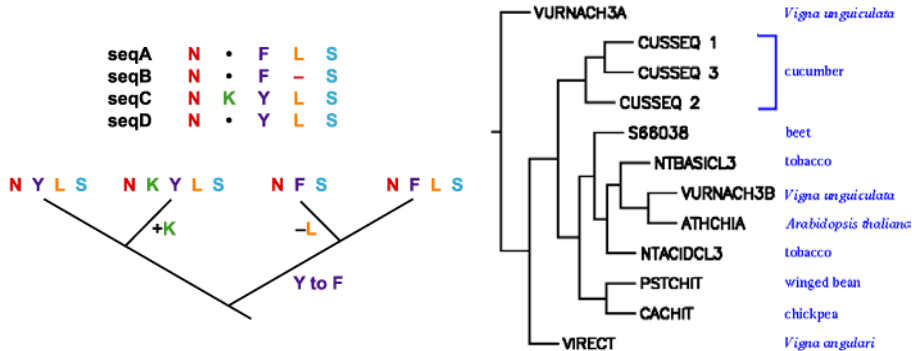


CMSC 838T – Lecture 5

◆ Phylogenetics

- Study of evolutionary relationships (sequences / species)
- Infer evolutionary relationship from shared features
- May improve multiple sequence alignment (MSA)



CMSC 838T – Lecture 5

Phylogenetics

◆ Phylogeny

- Relationship between organisms with common ancestor

◆ Phylogenetic tree

- Graph representing evolutionary history of sequence / species

◆ Premise

- Members sharing common evolutionary history (i.e., common ancestor) are more related to each other
- Can infer evolutionary relationship from shared features

◆ Long history of phylogenetics (from field of genetics)

- Historically → based on analysis of observable features (e.g., morphology, behavior, geographical distribution)
- Now → mostly analysis of DNA / RNA / amino acid sequences

CMSC 838T – Lecture 5

Phylogenetics – Motivation & Alignment

- ◆ **Goal of phylogenetics**
 - Understand relationship of sequence to similar sequences
 - Construct phylogenetic tree representing evolutionary history
- ◆ **Motivation / application**
 - Identify closely related families
 - Use phylogenetic relationships to predict gene function
 - Follow changes in rapidly evolving species (e.g., viruses)
 - Analysis can reveal which genes are under selection
 - Provide epidemiology for tracking infections & vectors
 - Few direct applications
- ◆ **Relationship to multiple sequence alignment (MSA)**
 - Alignment of sequences should take evolution into account
 - More precise phylogenetic relationships ↔ improved MSA

CMSC 838T – Lecture 5

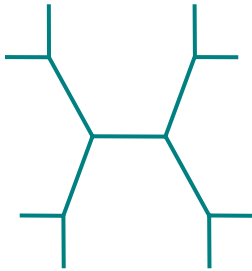
Phylogenetics Overview

- ◆ **Phylogenetic trees**
- ◆ **Tree construction algorithms**
 - Distance methods
 - UPGMA
 - Neighbor-joining
 - Maximum parsimony
 - Maximum likelihood
- ◆ **Assessing phylogenetic trees**

CMSC 838T – Lecture 5

Phylogenetic Trees

Unrooted tree
(Dendrogram)



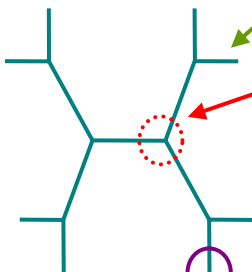
Rooted trees



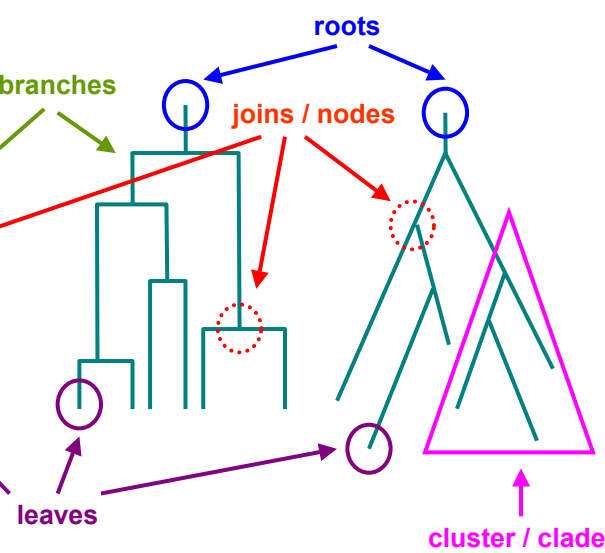
CMSC 838T – Lecture 5

Phylogenetic Trees

Unrooted tree
(Dendrogram)



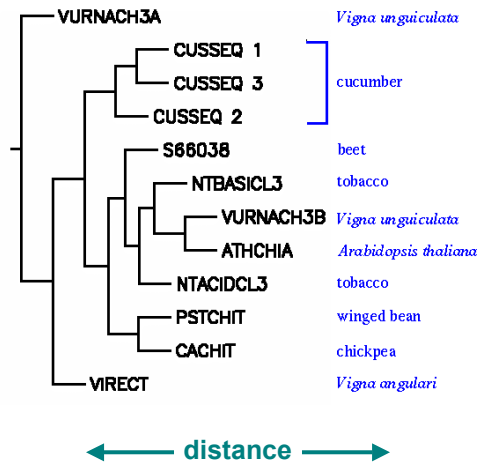
Rooted trees



CMSC 838T – Lecture 5

Phylogenetic Trees

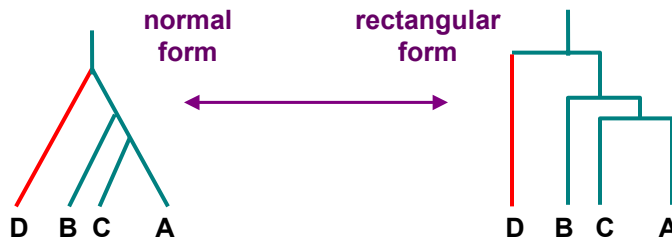
- ◆ **Leaves / taxa**
 - Original sequences
- ◆ **Branches**
 - Represent change
 - Length represents evolutionary **distance**
- ◆ **Cluster / clade**
 - All sequences in subtree with common ancestor (treated as single node)
- ◆ **Join / node**
 - Point of joining two leaves / clusters



CMSC 838T – Lecture 5

Phylogenetic Trees

- ◆ **Use binary trees (evolution is bifurcating process)**
 - Can approximate all tree shapes (w/ arbitrarily short edges)
 - Simplifies tree generation & analysis
- ◆ **Trees can be represented in rectangular form**
 - Alternative form of representation
 - Distance determined only by “height” of branch



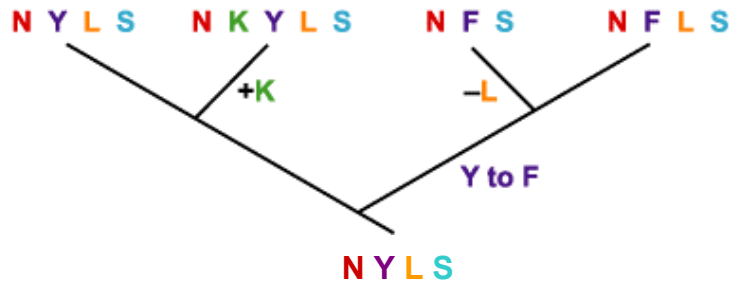
CMSC 838T – Lecture 5

Phylogenetic Trees

- ◆ Can label branches of tree with change to sequence

```

seqA  N • F L S
seqB  N • F - S
seqC  N K Y L S
seqD  N • Y L S
    
```



CMSC 838T – Lecture 5

Phylogenetic Trees – Distance

- ◆ (Evolutionary) Distance

- Many possible measures
 - Fraction of sites that differ between two sequences
 - # of changes needed to convert one sequence to another
 - Pairwise alignment scores, normalized by average score for random alignment [Feng & Doolittle 1996]
- $$\text{Score} = (\text{S.actual} - \text{S.random}) / (\text{S.identical} - \text{S.random})$$
 Where s.identical = score for aligning identical sequence

- ◆ Distance matrix

- Matrix of pairwise distances between all sequences
- Used to generate tree

Seq.	A	B	C	D
A	—	8	7	12
B		—	9	14
C			—	11
D				—

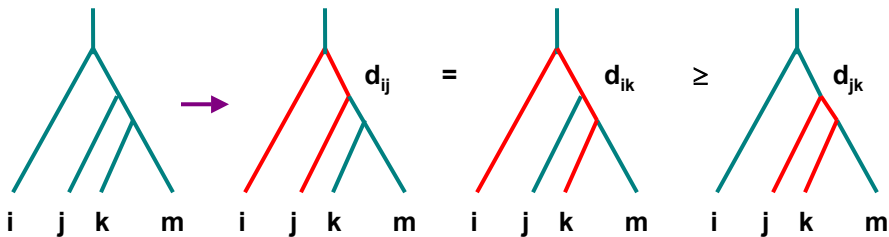
- ◆ Tree shape

- Varies with construction method, distance metric

CMSC 838T – Lecture 5

Phylogenetic Trees – Distance

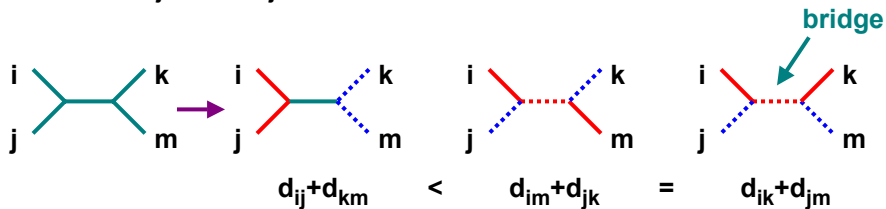
- ◆ Distances are **ultrametric** if
 - Same rate of change on all branches in tree (rare in practice)
 - All leaves equidistant from root
 - Also known as a “**molecular clock**”
 - Distance matrix must satisfy the following **3-point condition**
 - For any three leaves i, j, k , distances d_{ij}, d_{ik}, d_{jk}
 - two of three distances are equal and \geq third



CMSC 838T – Lecture 5

Phylogenetic Trees – Distance

- ◆ Distances are **additive** if
 - Distance between any two leaves i & j on tree = sum of lengths of edges connecting i & j
 - Distance matrix must satisfy the following **4-point condition**
 - For any four leaves i, j, k, m , two of the distances $d_{ij}+d_{km}, d_{ik}+d_{jm}, d_{im}+d_{jk}$ are equal and greater than the third



- In fact, the difference is $2 \times$ the length of the “**bridge**” edge(s)

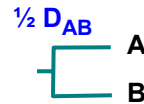
CMSC 838T – Lecture 5

Tree Construction – UPGMA

- ◆ **UPGMA (Unweighted Pair Group Method using Arithmetic Averages)** [Sokal & Michener 1958]

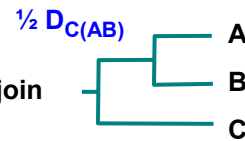
- ◆ **Algorithm**

1. Find pair of sequences A, B with smallest distance D_{AB}
2. Insert join for A, B at tree height = $\frac{1}{2} D_{AB}$
3. Update distance to new cluster as the average distance between pairs of sequences in each cluster
4. Repeat until all sequences / clusters joined
5. Produces rooted tree



- ◆ **Assumptions**

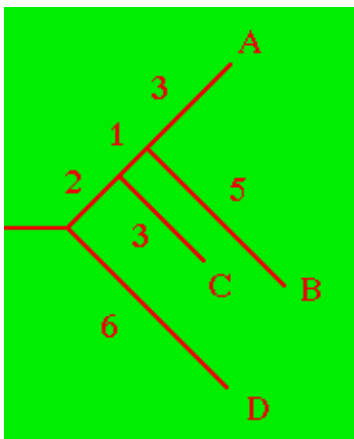
- Distances for tree are **ultrametric**
 - Branch lengths for 2 leaves same after join
- Distances for tree are **additive**



CMSC 838T – Lecture 5

Tree Construction Example

Original tree



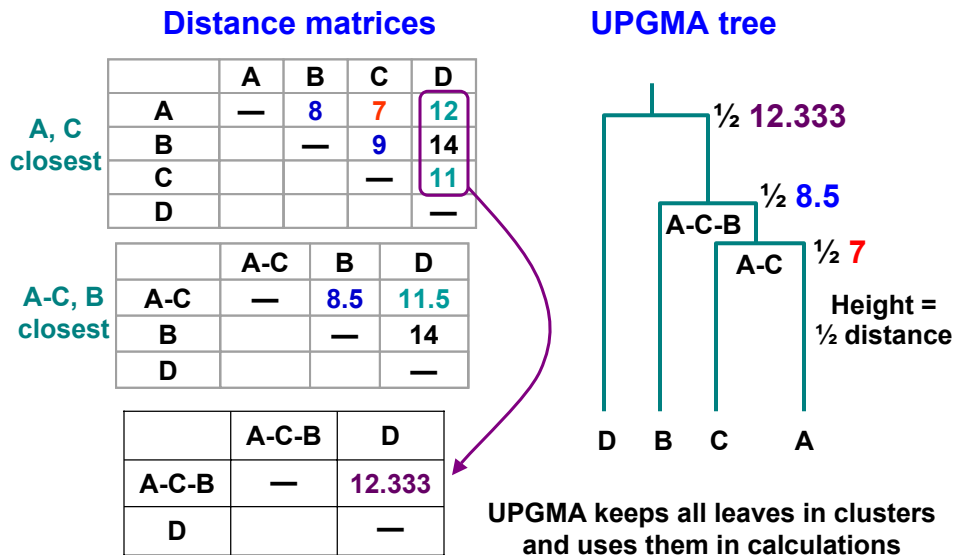
Distance matrix

Sequences	A	B	C	D
A	—	8	7	12
B		—	9	14
C			—	11
D				—

Note that tree distances are **additive** (i.e., distance between X, Y = sum of lengths of edges connecting X, Y)

CMSC 838T – Lecture 5

Tree Construction Example – UPGMA



CMSC 838T – Lecture 5

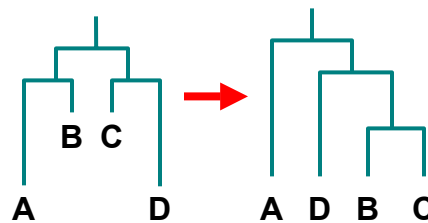
Tree Construction – Neighbor-Joining

◆ Goal

- Join closest **neighbors** (nodes w / same parent) in tree
- Avoids problem with UPGMA when rates of change differ

◆ Example

- Closest leaves not neighbors in correct tree, but joined first by UPGMA



◆ Assumptions

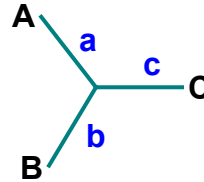
- Rate of change can differ
 - Branch lengths may differ after join
- Branch lengths for tree are **additive**

CMSC 838T – Lecture 5

Neighbor-Joining – Basic Principle

- ◆ Calculating branch lengths after join (additive tree)

	A	B	C
A	—	$d_{A,B}$	$d_{A,C}$
B		—	$d_{B,C}$
C			—



- ◆ Simple algebra shows

- Given
 - $d_{A,B} = a + b$
 - $d_{A,C} = a + c$
 - $d_{B,C} = b + c$
- We can calculate
 - $a = \frac{1}{2} (d_{A,B} + d_{A,C} - d_{B,C})$
 - $b = \frac{1}{2} (d_{A,B} + d_{B,C} - d_{A,C})$
 - $c = \frac{1}{2} (d_{B,C} + d_{A,C} - d_{A,B})$

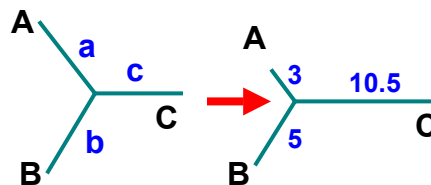
CMSC 838T – Lecture 5

Neighbor-Joining – Basic Principle

- ◆ Example (additive tree, not ultrametric)

- Given distance matrix, calculate branch lengths

	A	B	C
A	—	8	13.5
B		—	15.5
C			—



Calculation results

$$a = \frac{1}{2} (d_{A,B} + d_{A,C} - d_{B,C}) = \frac{1}{2} (8 + 13.5 - 15.5) = 3$$

$$b = \frac{1}{2} (d_{A,B} + d_{B,C} - d_{A,C}) = \frac{1}{2} (8 + 15.5 - 13.5) = 5$$

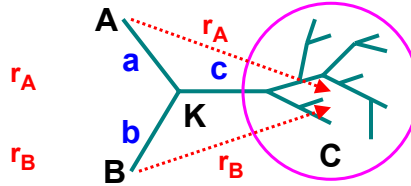
$$c = \frac{1}{2} (d_{B,C} + d_{A,C} - d_{A,B}) = \frac{1}{2} (15.5 + 13.5 - 8) = 10.5$$

CMSC 838T – Lecture 5

Neighbor-Joining – Basic Principle

- ◆ Exploit principle for neighbor-joining algorithm

	A	B	C
A	—	$d_{A,B}$	$d_{A,C}$
B		—	$d_{B,C}$
C			—



Simply treat all other nodes as C, and treat distance to C as r

- ◆ Replace distance to C

- Used **normalized divergence** r_A r_B (~ avg. distance to nodes)
- We can calculate
 - $a = \frac{1}{2} (d_{A,B} + d_{A,C} - d_{B,C}) \rightarrow \frac{1}{2} (d_{A,B} + r_A - r_B)$
 - $b = \frac{1}{2} (d_{A,B} + d_{B,C} - d_{A,C}) \rightarrow \frac{1}{2} (d_{A,B} + r_B - r_A)$
 - $c = \frac{1}{2} (d_{B,C} + d_{A,C} - d_{A,B}) \rightarrow \frac{1}{2} (d_{B,C} + d_{A,C} - d_{A,B})$

CMSC 838T – Lecture 5

Tree Construction – Neighbor-Joining

- ◆ Approach

- To find closest pair of neighbors
 - Reduce branch length for a node by (approximately) the average distance of the node from all other nodes
 - Find smallest distance between nodes (after reduction)

- ◆ Definitions

For all pairs of nodes **A** & **B** in set of all nodes **L**, let

$d_{A,B}$ = distance between A,B

$R_X = \sum d_{X,N}$ where $N \in L$ (**total distance** from X to all N)

$r_X = R_X / (|L| - 2)$, where $|L| = \#$ of nodes

(**normalized divergence** from X to all other nodes)

$D_{A,B} = d_{A,B} - (r_A + r_B)$ (**rate-corrected distance**)

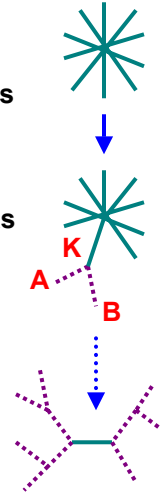
- ◆ **Key property** – 2 nodes w/ minimum **D** are always neighbors!

CMSC 838T – Lecture 5

Tree Construction – Neighbor-Joining

◆ Algorithm [Saitou & Nei 1987, Studier & Keppler 1988]

1. Begin with star tree & all sequences as nodes in L
2. Find pair of nodes **A** & **B** $\in L$ with minimum $D_{A,B}$
3. Create & insert new join (node **K**) w/ branch lengths
 - $d_{A,K} = \frac{1}{2} (d_{A,B} + r_A - r_B)$
 - $d_{B,K} = \frac{1}{2} (d_{A,B} + r_B - r_A)$
4. For remaining nodes $C \in L$, update distance to K as
 - $d_{K,C} = \frac{1}{2} (d_{A,C} + d_{B,C} - d_{A,B})$
5. Insert K and remove A, B from L
6. Repeat steps 2–5 until only two nodes left



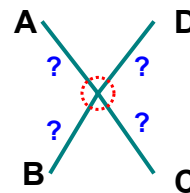
CMSC 838T – Lecture 5

Tree Construction Example – Neighbor Joining

(Rate-corrected) distance matrix

	A	B	C	D	R
A	—	8	7	12	13.5
B	-21	—	9	14	15.5
C	-20	-20	—	11	13.5
D	-20	-20	-21	—	18.5

normalized divergence =
 $\frac{\sum d}{(|L| - 2)}$
 $= \frac{\sum d}{2}$



Rate-corrected distances

$$\begin{aligned}
 D_{A,B} &= d_{A,B} - (r_A + r_B) &= 8 - (13.5 + 15.5) &= -21 \\
 D_{A,C} &= d_{A,C} - (r_A + r_C) &= 7 - (13.5 + 13.5) &= -20 \\
 D_{A,D} &= d_{A,D} - (r_A + r_D) &= 12 - (13.5 + 18.5) &= -20 \\
 D_{B,C} &= d_{B,C} - (r_B + r_C) &= 9 - (15.5 + 13.5) &= -20 \\
 D_{B,D} &= d_{B,D} - (r_B + r_D) &= 14 - (15.5 + 18.5) &= -20 \\
 D_{C,D} &= d_{C,D} - (r_C + r_D) &= 11 - (13.5 + 18.5) &= -21
 \end{aligned}$$

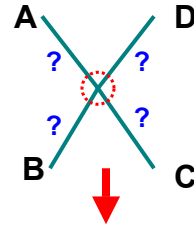
CMSC 838T – Lecture 5

Tree Construction Example – Neighbor Joining

(Rate-corrected) distance matrix

	A	B	C	D	r
A	—	8	7	12	13.5
B	-21	—	9	14	15.5
C	-20	-20	—	11	13.5
D	-20	-20	-21	—	18.5

normalized divergence =
 $\frac{\sum d}{(|L|-2)}$
 $= \frac{\sum d}{2}$



Edge lengths for A,B

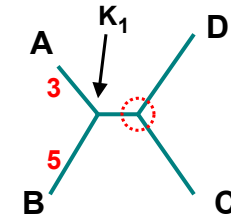
$$d_{A,K_1} = \frac{1}{2} (d_{A,B} + r_A - r_B) = \frac{1}{2} (8 + 13.5 - 15.5) = 3$$

$$d_{B,K_1} = \frac{1}{2} (d_{A,B} + r_B - r_A) = \frac{1}{2} (8 + 15.5 - 13.5) = 5$$

Distances to K_1

$$d_{K_1,C} = \frac{1}{2} (d_{A,C} + d_{B,C} - d_{A,B}) = \frac{1}{2} (7 + 9 - 8) = 4$$

$$d_{K_1,D} = \frac{1}{2} (d_{A,D} + d_{B,D} - d_{A,B}) = \frac{1}{2} (12 + 14 - 8) = 9$$



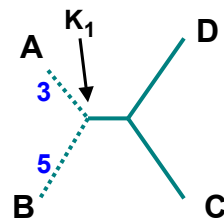
CMSC 838T – Lecture 5

Tree Construction Example – Neighbor Joining

(Rate-corrected) distance matrix

	C	D	K_1	r
C	—	11	4	15
D	-24	—	9	20
K_1	-24	-24	—	13

normalized divergence =
 $\frac{\sum d}{(|L|-2)}$
 $= \frac{\sum d}{1}$



Rate-corrected distances

$$D_{C,D} = d_{C,D} - (r_C + r_D) = 11 - (15 + 20) = -24$$

$$D_{C,K_1} = d_{C,K_1} - (r_C + r_{K_1}) = 4 - (15 + 13) = -24$$

$$D_{D,K_1} = d_{D,K_1} - (r_D + r_{K_1}) = 9 - (20 + 13) = -24$$

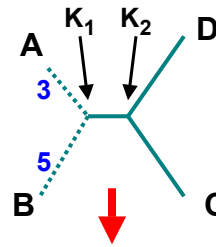
CMSC 838T – Lecture 5

Tree Construction Example – Neighbor Joining

(Rate-corrected) distance matrix

	C	D	K ₁	r
C	—	11	4	15
D	-24	—	9	20
K ₁	-24	-24	—	13

averaged distance =
 $\frac{\sum d}{(|L|-2)}$
 $= \frac{\sum d}{1}$



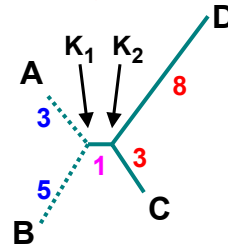
Edge lengths for C,D

$$d_{C,K_2} = \frac{1}{2} (d_{C,D} + r_C - r_D) = \frac{1}{2} (11 + 15 - 20) = 3$$

$$d_{D,K_2} = \frac{1}{2} (d_{C,D} + r_D - r_C) = \frac{1}{2} (11 + 20 - 15) = 8$$

Distances to K₂

$$d_{K_2,K_1} = \frac{1}{2} (d_{K_1,C} + d_{K_1,D} - d_{C,D}) = \frac{1}{2} (4 + 9 - 11) = 1$$



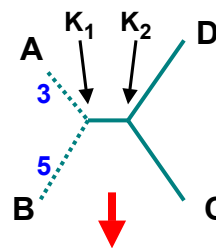
CMSC 838T – Lecture 5

Tree Construction Example – Neighbor Joining

(Rate-corrected) distance matrix

	C	D	K ₁	r
C	—	11	4	15
D	-24	—	9	20
K ₁	-24	-24	—	13

averaged distance =
 $\frac{\sum d}{(|L|-2)}$
 $= \frac{\sum d}{1}$



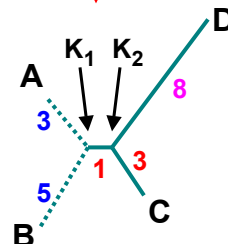
Edge lengths for C, K₁

$$d_{C,K_2} = \frac{1}{2} (d_{C,K_1} + r_C - r_{K_1}) = \frac{1}{2} (4 + 15 - 13) = 3$$

$$d_{K_1,K_2} = \frac{1}{2} (d_{C,K_1} + r_{K_1} - r_C) = \frac{1}{2} (4 + 13 - 15) = 1$$

Distances to K₂

$$d_{K_2,D} = \frac{1}{2} (d_{D,C} + d_{D,K_1} - d_{C,K_1}) = \frac{1}{2} (11 + 9 - 4) = 8$$



CMSC 838T – Lecture 5

Tree Construction Example – Neighbor Joining

(Rate-corrected) distance matrix

	C	D	K ₁	r
C	—	11	4	15
D	-24	—	9	20
K ₁	-24	-24	—	13

averaged distance =
 $\frac{\sum d}{(|L|-2)}$
 $= \frac{\sum d}{1}$

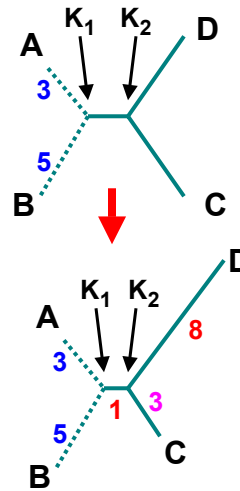
Edge lengths for D, K₁

$$d_{D,K_2} = \frac{1}{2} (d_{D,K_1} + r_D - r_{K_1}) = \frac{1}{2} (9 + 20 - 13) = 8$$

$$d_{K_1,K_2} = \frac{1}{2} (d_{D,K_1} + r_{K_1} - r_D) = \frac{1}{2} (9 + 13 - 20) = 1$$

Distances to K₂

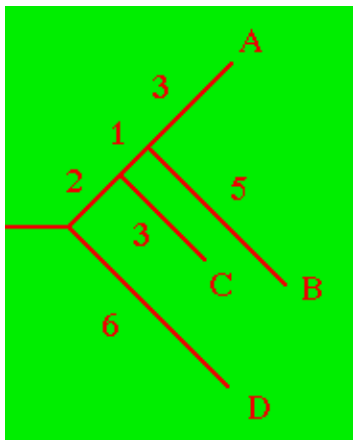
$$d_{K_2,C} = \frac{1}{2} (d_{C,D} + d_{C,K_1} - d_{D,K_1}) = \frac{1}{2} (11 + 4 - 9) = 3$$



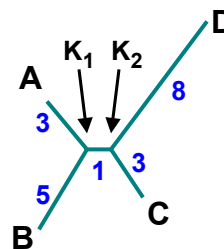
CMSC 838T – Lecture 5

Tree Construction Example – Neighbor Joining

Original tree



Neighbor-joining tree



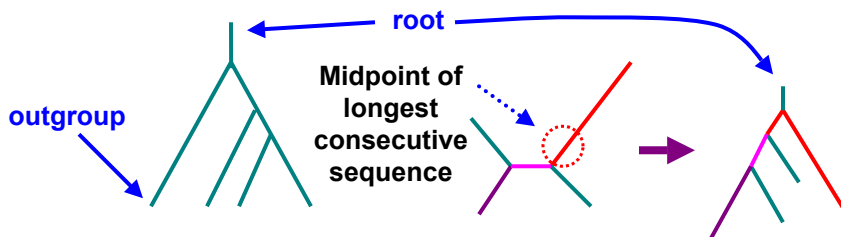
Except for missing root,
 finds same tree topology

CMSC 838T – Lecture 5

Tree Construction – Distance Methods

◆ Inserting root

- Neighbor-joining terminates w/ 2 nodes, outputs unrooted tree
- If need to select root
 - Find **outgroup** node known to be more distant, insert root nearby, or
 - Find longest consecutive sequence of edges, insert root near middle (assumes evolution rates comparable)



CMSC 838T – Lecture 5

Tree Construction – Distance Methods

◆ Complexity

- Distance-based methods much faster than other methods
- Commonly used in multiple sequence alignment
 - UPGMA – PILEUP
 - Neighbor-joining – CLUSTALW

◆ Problems

- Both UPGMA & neighbor-joining are greedy heuristics
- Possible to be trapped in local maxima (no backtracking)
- Output is a single tree, even if many equal-cost alternatives

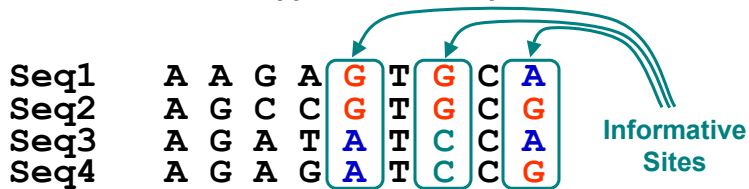
◆ May use as starting point

- Tree generated provides upper bound for branch-and-bound
- Initial tree for probabilistic branch-swapping techniques

CMSC 838T – Lecture 5

Tree Construction – Maximum Parsimony

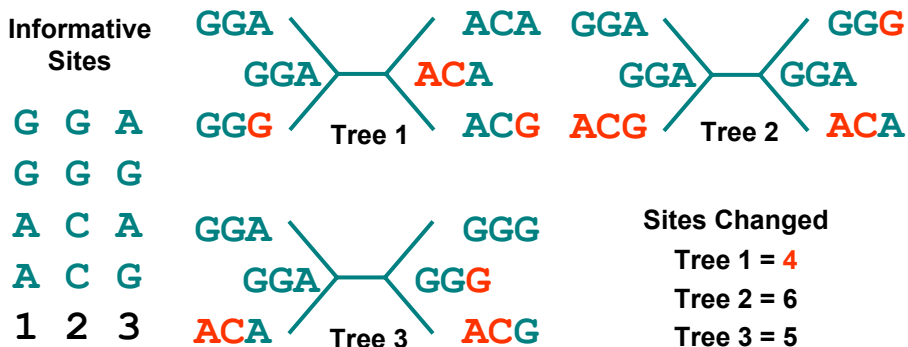
- ◆ **Maximum parsimony** [Fitch 1971]
 - Minimize number of sequence changes in tree
 - Assume fewest changes (mutations) = most likely (evolution)
- ◆ **Informative site**
 - Position with useful change information (for parsimony)
 - I.e., # of changes in position dependent on tree chosen
 - Must have ≥ 2 different bases / residues, such that each base / residue appears in ≥ 2 sequences



CMSC 838T – Lecture 5

Tree Construction – Maximum Parsimony

- ◆ **Most parsimonious tree**
 - Tree with fewest total # of changes at informative sites



CMSC 838T – Lecture 5

Tree Construction – Maximum Parsimony

◆ Algorithm

- Generate all possible tree topologies
- Count number of changes required
- Select tree with minimum # changes
- Use branch-and-bound to reduce search
 - Search trees with increasing # of leaves
 - Abandon subtree when # changes \geq best completed tree

◆ Characteristics

- Computationally expensive
- Analyze only informative sites
- Misleading if rates of changes vary among branches
- Evolution is not always parsimonious

CMSC 838T – Lecture 5

Tree Construction – Maximum Likelihood

◆ Goal

- Given the probability $P(x|y,t)$ for a sequence y to evolve (mutate) to sequence x along an edge of length t (time)
- Find tree that has highest probability of taking place

◆ Mutation probabilities

- Bases: Jukes-Cantor model [Jukes-Cantor 1969, Kimura 1980]
- Amino acids: PAM [Dayhoff+ 1978]

◆ Algorithm

- Search over all tree topologies & sequence assignments
- For each topology & assignment, search all branch lengths

◆ Characteristics

- Very computationally expensive

CMSC 838T – Lecture 5

Tree Construction – Issues

- ◆ **Selecting tree construction algorithm**
 - If strong sequence similarity → maximum parsimony
 - If clearly recognizable sequence similarity → distance methods
 - Otherwise → maximum likelihood
- ◆ **Determining statistical significance**
 - Multiple tree shapes possible
 - Find probability that tree shape is as described
 - Sample by “bootstrapping” [Efron & Tibshirani 1993]
 - Generate artificial data set by repeatedly selecting random columns of alignment (**pseudo-alignment**) with replacement
 - Build tree for pseudo-alignments many (1000+) times
 - Frequency phylogenetic feature appears → confidence level

CMSC 838T – Lecture 5

Phylogenetics – Issues

- ◆ **Gene trees vs species trees**
 - Gene duplication can complicate phylogenetic analysis
 - Paralogues (duplicated genes) do not fit in evolutionary tree
- ◆ **Choice of target sequence type**
 - Ribosomal RNA (slowest change / mutation rate)
 - Use for very long-term evolutionary studies, spanning species boundaries & biological kingdoms
 - DNA / RNA (fastest change / mutation rate)
 - Use for short-term studies of closely-related species
 - Contains more evolutionary information than protein
 - Protein (medium change / mutation rate)
 - Use for wide species comparisons
 - More reliable alignment than DNA

CMSC 838T – Lecture 5

Plylogenetics Summary

- ◆ **Phylogenetic prediction**
 - Infer evolutionary relationships from shared features
 - May have application to sequence alignment, epidemiology
- ◆ **Phylogenetic trees**
 - May be ultrametric and / or additive
- ◆ **Tree construction**
 - Inexpensive distance-based (UPGMA, neighbor-joining)
 - Expensive (exhaustive) tree searches (parsimony, likelihood)
- ◆ **Assessing phylogenetic trees**
 - Algorithms always produce some tree (of varying accuracy)
 - Expert biology knowledge to assess correctness / significance

CMSC 838T – Lecture 5

Where Are We Now?

- ◆ **Bioinformatics topics covered**
 - Molecular biology background
 - Pairwise sequence alignment
 - Multiple sequence alignment
 - Phylogenetics
- ◆ **Remaining bioinformatics topics**
 - Protein structure prediction
 - Gene assembly and prediction
 - Microarrays & expressed sequence tag (EST) analysis
 - Sequence / structure database search & organization
- ◆ **High performance computing...**

CMSC 838T – Lecture 5

More Bioinformatics Terms

- ◆ **Functional genomics**
 - Identify function of genes in organism
- ◆ **Comparative genomics**
 - Identify genes
 - Related to other genes in organism
 - Related to genes in other species
 - Create evolutionary history of related genes
 - Locate insertions, deletions, substitutions occurring in evolution
- ◆ **Proteomics**
 - Identify & characterize all gene products (proteins) in organism
- ◆ **Structural proteomics**
 - Identify or predict 3D structure of all proteins in organism

CMSC 838T – Lecture 5

More Bioinformatics Terms

- ◆ **Pharmacogenomics**
 - Application of genomic approaches to identify drug targets
 - Searching genomes for potential drug receptors
 - Examining characteristic gene expression in pathogens & hosts during infection for diagnostics or therapy targets
 - Cataloguing & processing info on pharmacology & genetics
- ◆ **Pharmacogenetics**
 - Identifying genetic causes for individualized drug responses
 - Identify genetic variation (e.g., SNPs) characteristic of particular patient response profiles
 - Use to improve administration & development of therapies
 - Identify receptive patient subsets, optimize drug dosages
- ◆ **Lots of data mining...**

CMSC 838T – Lecture 5

More Bioinformatics Terms

◆ Medical informatics

- Techniques to improve usefulness & management of medical information
- Emphasis on structures and algorithms for the manipulation of medical data, rather than understanding the data itself
- Mostly databases & data integration, little bioinformatics

CMSC 838T – Lecture 5

Popular Bioinformatics Resources

◆ Software tools

- Sequence search – BLAST
- Sequence analysis – EMBOSS, Staden
- Structure prediction – THREADER, PHD
- Molecular imaging / modeling – RasMol, WHATIF

◆ Public databases

- Nucleotides – GenBank
- Proteins – Protein DataBank (PDB)
- Biological papers – PubMed

CMSC 838T – Lecture 5