

# Phylogenetic Trees

## Distance trees

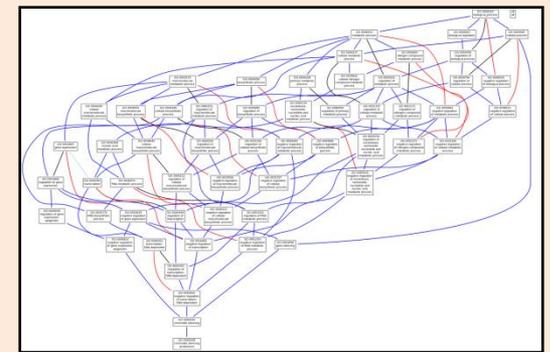
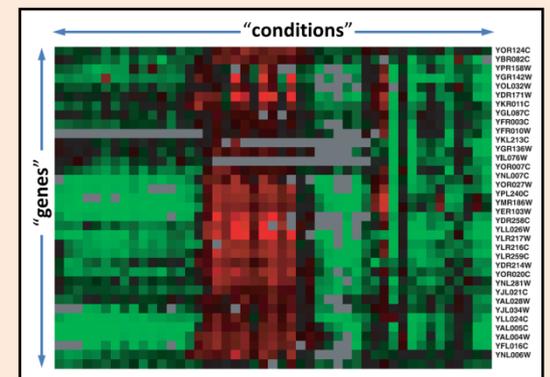
Genome 373

Genomic Informatics

Elhanan Borenstein

# A quick review

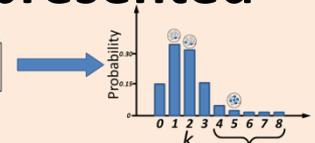
- Gene expression profiling
  - Which molecular processes/functions are involved in a certain phenotype (e.g., disease, stress response, etc.)
- The **Gene Ontology (GO) Project**
  - Provides shared vocabulary/annotation
  - GO terms are linked in a complex structure
- **Enrichment analysis:**
  - Find the “most” differentially expressed genes
  - Identify functional annotations that are **over-represented**
  - Modified Fisher's exact test



Hypergeometric distribution

$$\mathbb{P}(\sigma_i = k) = \frac{\binom{m}{k} \binom{m-n}{n-k}}{\binom{m}{n}}$$

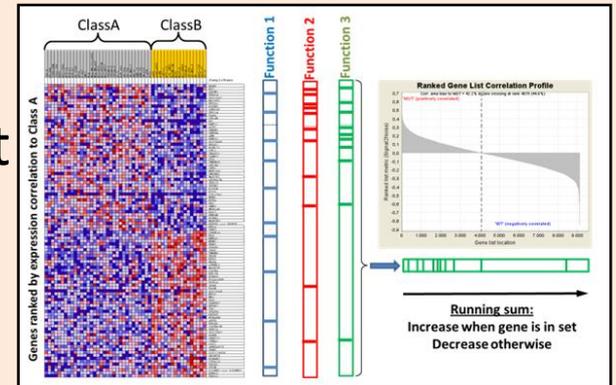
$m=50, m_1=10, n=8$



# A quick review – cont'

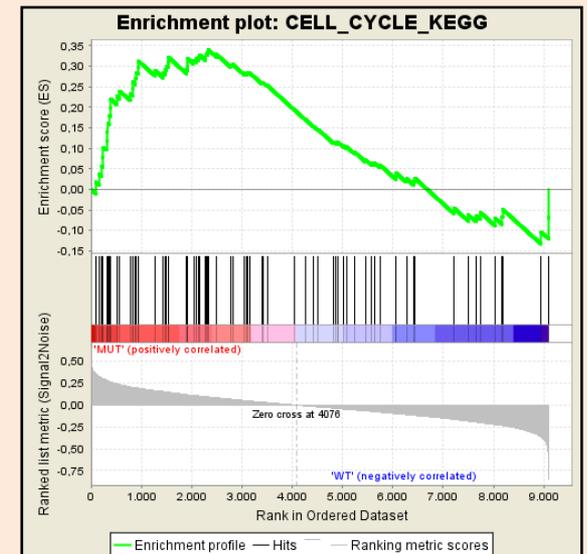
## ■ Gene Set Enrichment Analysis

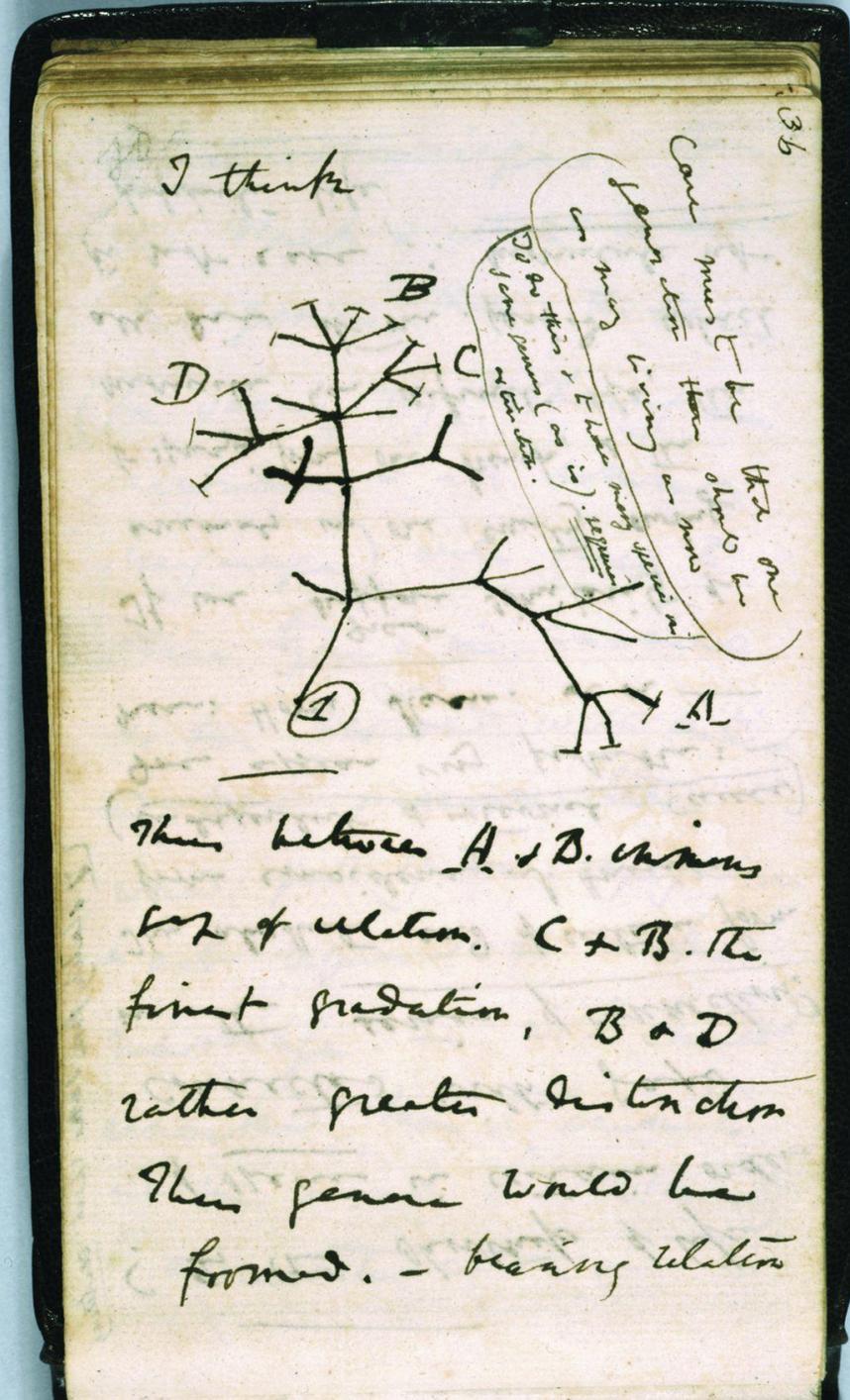
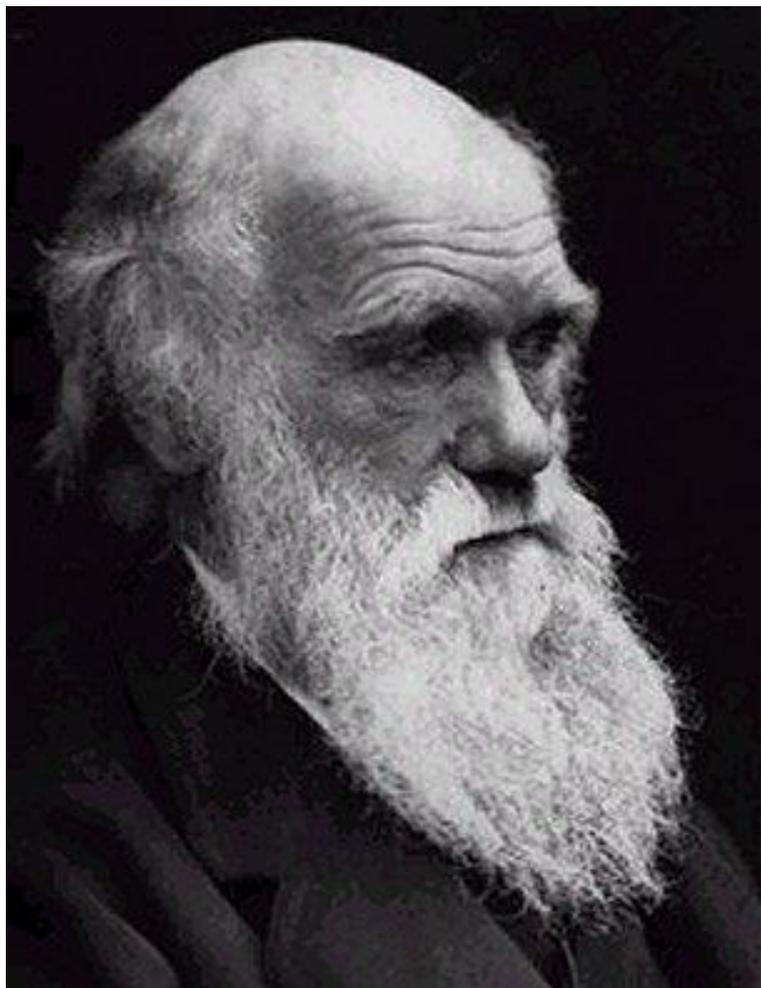
- Calculates a score for the enrichment of a entire set of genes
- Does not require setting a cutoff!
- Identifies the set of relevant genes!
- Provides a more robust statistical framework!



## ■ GSEA steps:

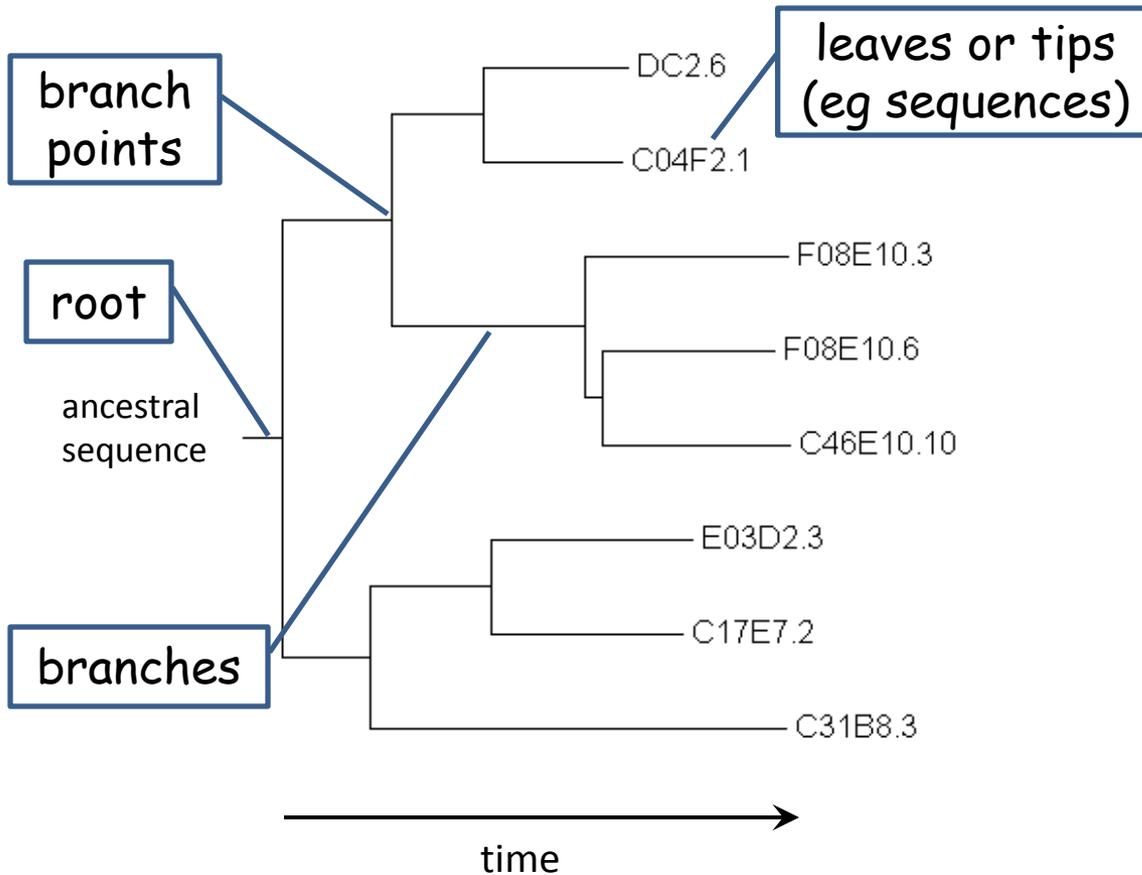
1. Calculation of an enrichment score (ES) for each functional category
2. Estimation of significance level
3. Adjustment for multiple hypotheses testing



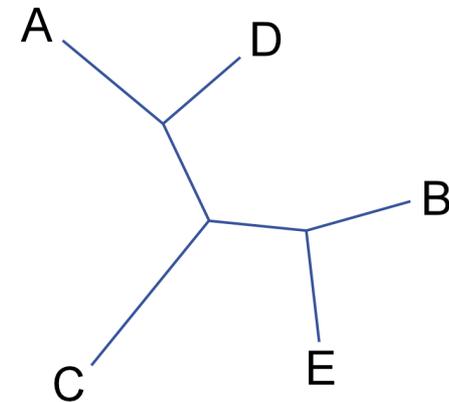


# Defining what a “tree” means

rooted tree (all real trees are rooted):



unrooted tree:  
(used when the root isn't known):

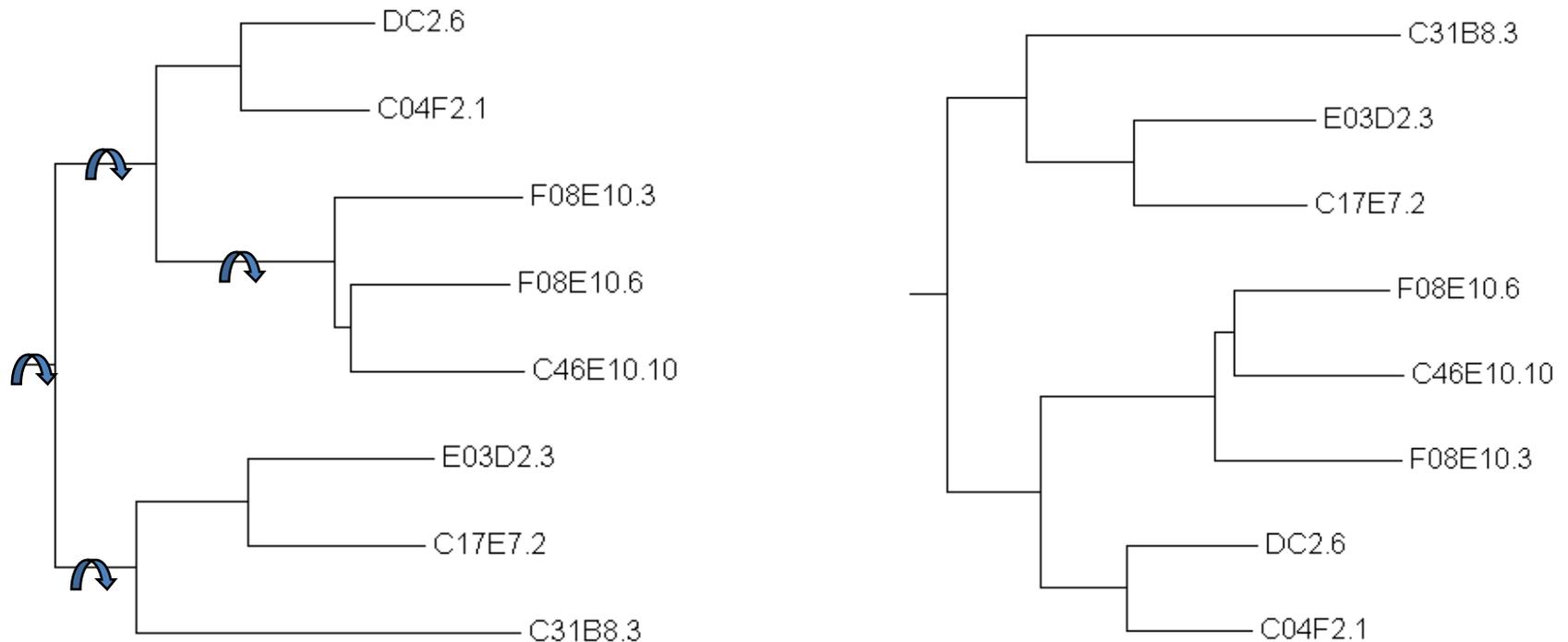


time radiates out from  
somewhere (probably  
near the center)

... sequence divergence is proportional to  
(horizontal) branch lengths

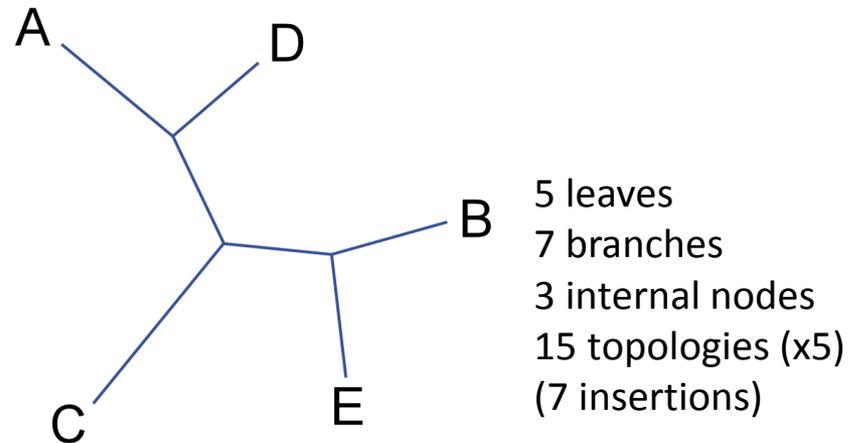
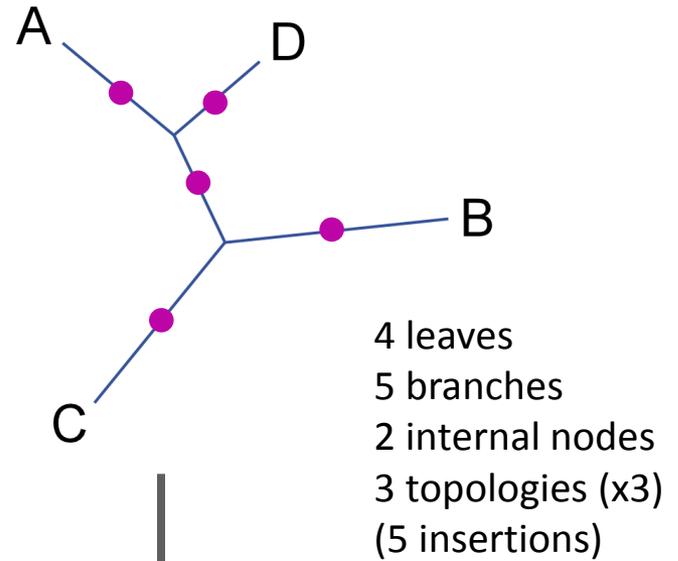
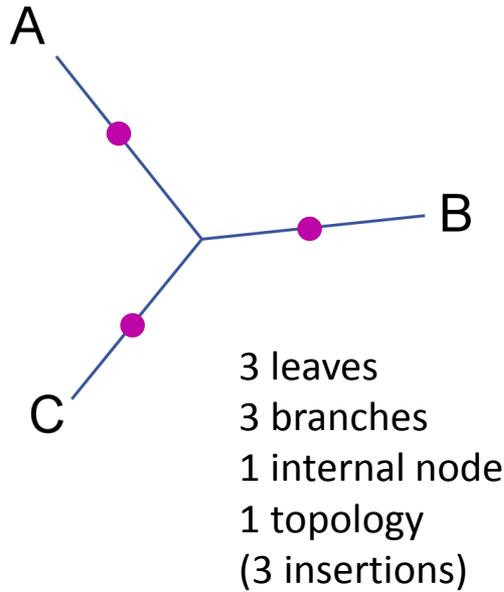
# A tree has topology and distances

Are these topologically different trees?



Topologically, these are the *SAME* tree. In general, two trees are the same if they can be inter-converted by branch rotations.

# The number of tree topologies grows extremely fast



In general, an unrooted tree with **N** leaves has:

**$2N - 3$**  total branches

**N leaf** branches

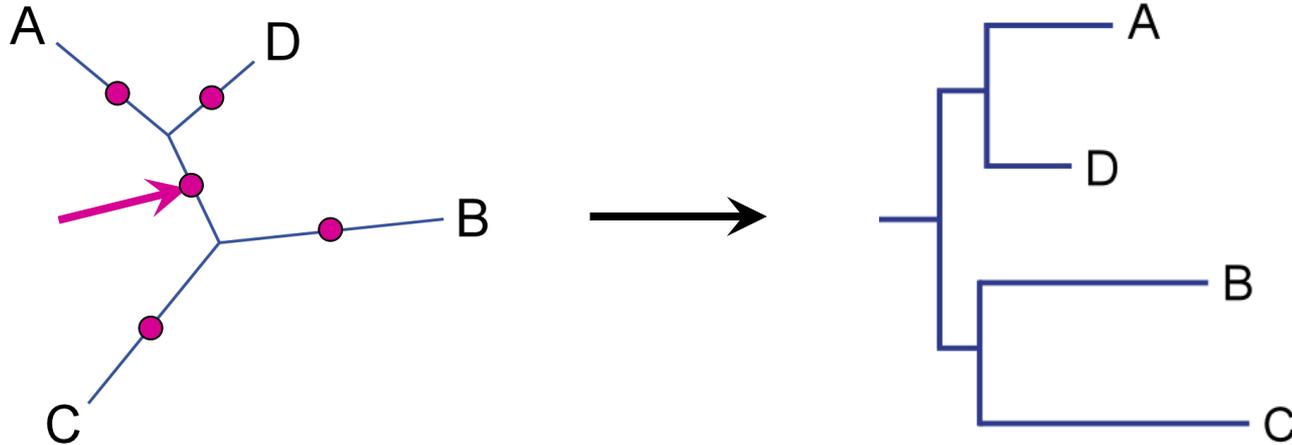
**$N - 3$**  internal branches

**$N - 2$**  internal nodes

**$3 \cdot 5 \cdot 7 \cdot \dots \cdot (2N - 5) \sim O(N!)$**  topologies

# There are many rooted trees for each unrooted tree

For each unrooted tree, there are  $2N - 3$  times as many rooted trees, where  $N$  is the number of leaves ( $\#$  internal branches =  $2N - 3$ ).



20 leaves - 564,480,989,588,730,591,336,960,000,000 topologies

# How can you compute a tree?

- Many methods available, we will talk about:
  - Distance trees
  - Parsimony trees
- Others include:
  - Maximum-likelihood trees
  - Bayesian trees

# Distance matrix methods

- Methods based on a set of **pairwise distances** typically from a multiple alignment.

	1	2	3	4	5	6
human	a	g	t	c	t	c
chimp	a	g	a	g	t	c
gorilla	c	g	g	c	a	g
orangutan	c	g	g	g	a	c

human - chimp has 2 changes out of 6 sites  
human - orang has 4 changes of out 6 sites  
etc.



	human	chimp	gorilla	orang
human	0	2/6	4/6	4/6
chimp		0	5/6	3/6
gorilla			0	2/6
orang				0

(symmetrical, lower left not filled in)

- Try to build the tree whose distances best match the real distances.

# Best Match?

- "Best match" based on least squares of real pairwise distances compared to the tree distances:

Let  $D_m$  be the measured distances.

Let  $D_t$  be the tree distances.

Find the tree that minimizes:

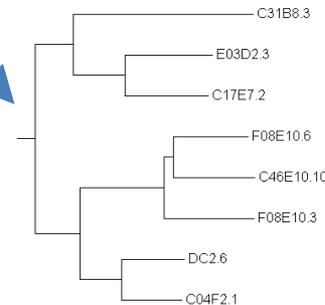
$$\sum_{i=1}^N (D_t - D_m)^2$$

	1	2	3	4	5	6
human	a	g	t	c	t	c
chimp	a	g	a	g	t	c
gorilla	c	g	g	c	a	g
orangutan	c	g	g	a	a	c

human - chimp has 2 changes out of 6 sites  
human - orang has 4 changes of out 6 sites  
etc.

	human	chimp	gorilla	orang
human	0	2/6	4/6	4/6
chimp		0	5/6	3/6
gorilla			0	2/6
orang				0

(symmetrical, lower left not filled in)



# Enumerate and score all trees?

- **How about the following algorithm:**  
Enumerate every tree topology, fit least-squares best distances for each topology, keep best.
- Not used for distance trees - there is a much faster way to get very close to correct.

# The UPGMA algorithm

- 1) generate a table of pairwise sequence distances and assign each sequence to a list of  $N$  tree nodes.
- 2) look through current list of nodes (initially these are all leaf nodes) for the pair with the smallest distance.
- 3) merge the closest pair, remove the pair of nodes from the list and add the merged node to the list.
- 4) repeat until only one node left in list - it is the root.

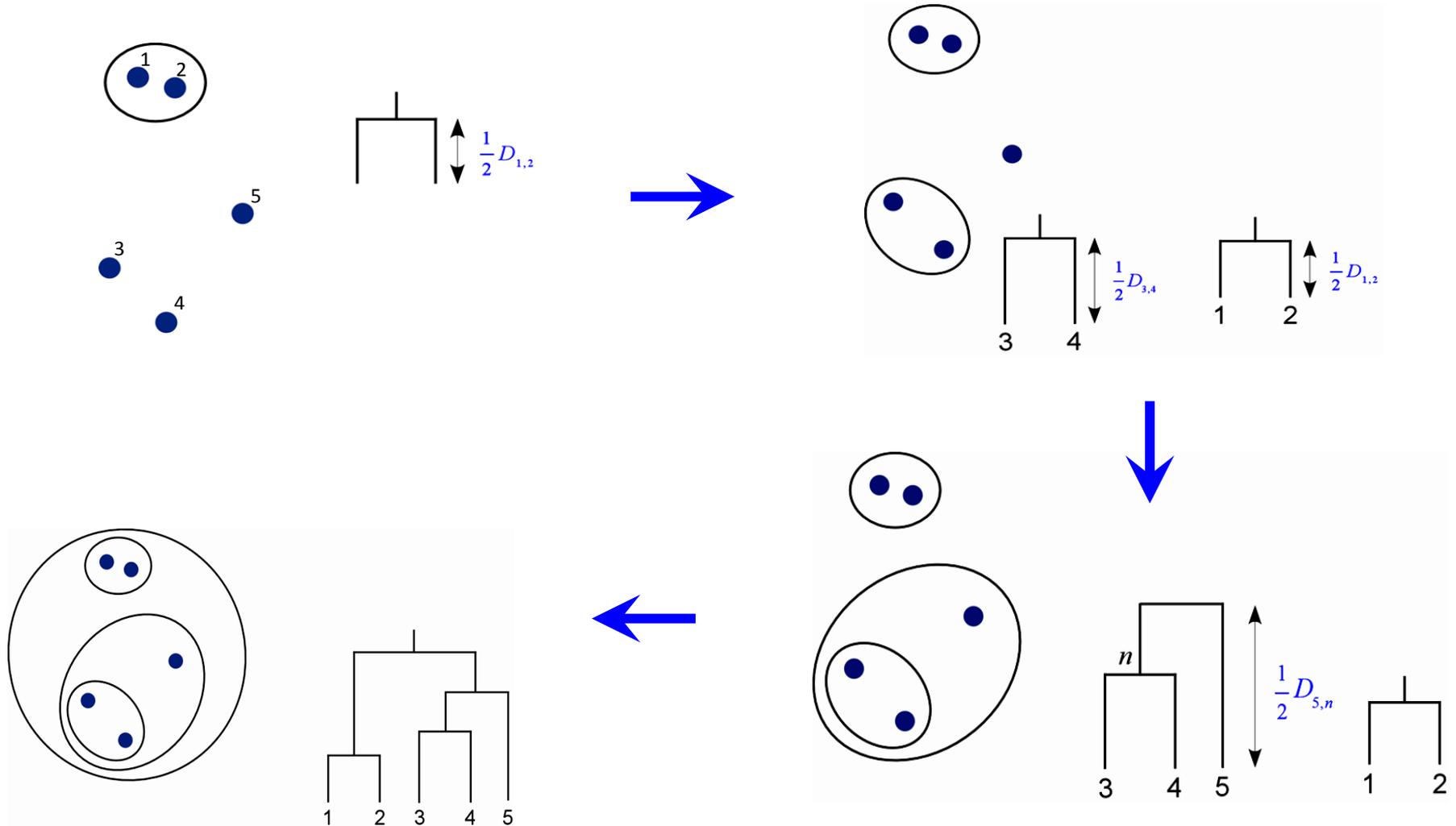
$$D_{n1,n2} = \frac{1}{N} \sum_i \sum_j d_{ij}$$

where  $i$  is each leaf of  $n1$  (node1),  $j$  is each leaf of  $n2$  (node2),  
and  $N$  is the number of distances summed

definition of  
distance

(in words, this is just the arithmetic average of the distances between all the leaves in one node and all the leaves in the other node)

# UPGMA (Unweighted Pair Group Method with Arithmetic Mean)



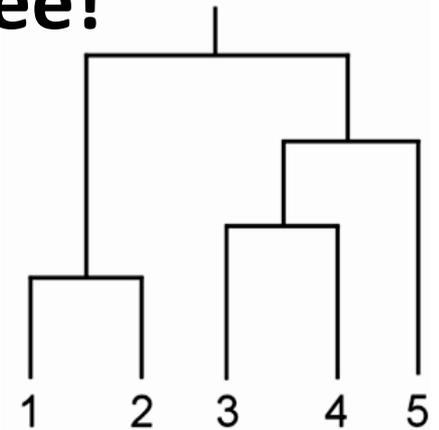
**mmm...**

**Déjà vu anyone?**

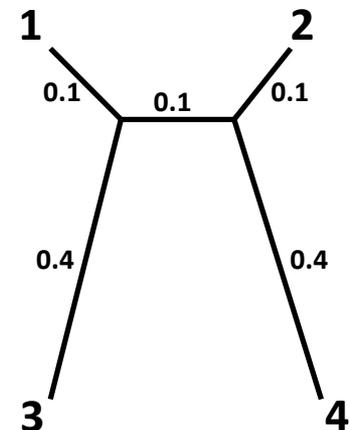
# The Molecular Clock

- **UPGMA assumes a constant rate of the molecular clock across the entire tree!**

- The sum of times down a path to any leaf is the same



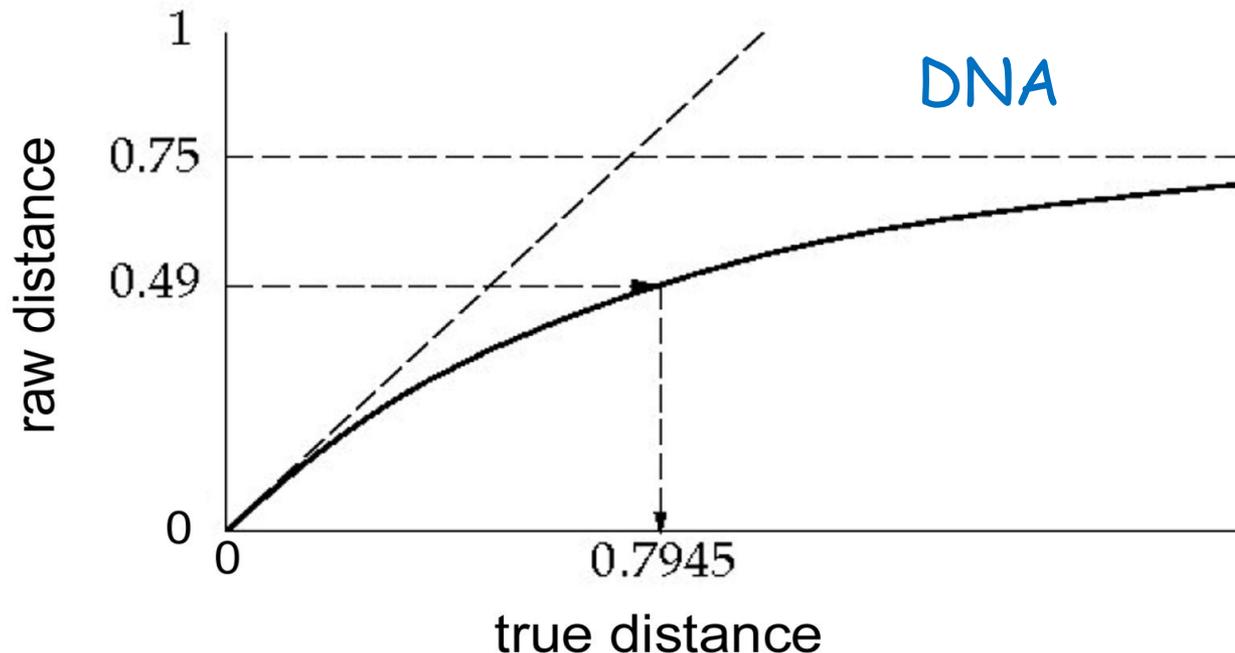
- This assumption may not be correct ... and will lead to incorrect tree reconstruction.





# Raw distance correction

- As two DNA sequences diverge, it is easy to see that their maximum raw distance is  $\sim 0.75$  (assuming equal nt frequencies,  $\frac{1}{4}$  of residues will be identical even if unrelated sequences).
- We would like to use the "true" distance, rather than raw distance.
- This graph shows evolutionary distance related to raw distance:



# Mutational models for DNA

- Jukes-Cantor (JC) - all mutations equally likely.
- Kimura 2-parameter (K2P) - transitions and transversions have separate rates.
- Generalized Time Reversible (GTR) - all changes have separate rates.

(Models similar to GTR are also available for protein)

# Jukes-Cantor model

Jukes-Cantor model:

$$D = -\frac{3}{4} \ln\left(1 - \frac{4}{3} D_{raw}\right)$$

$D_{raw}$  is the raw distance (what we directly measure)

$D$  is the corrected distance (what we want)

# Distance trees - summary

- Convert each pairwise raw distance to a corrected distance.
- Build tree as before (UPGMA algorithm).
- Notice that these methods don't need to consider all tree topologies - they are very fast, even for large trees.

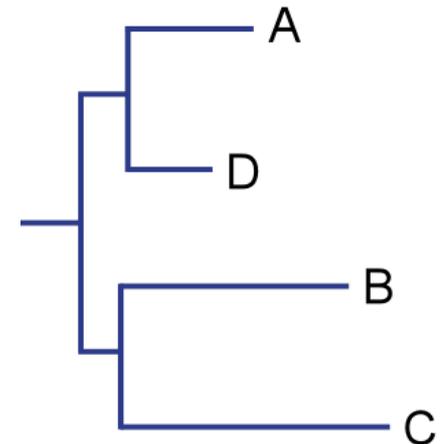


# Representing a tree in Python

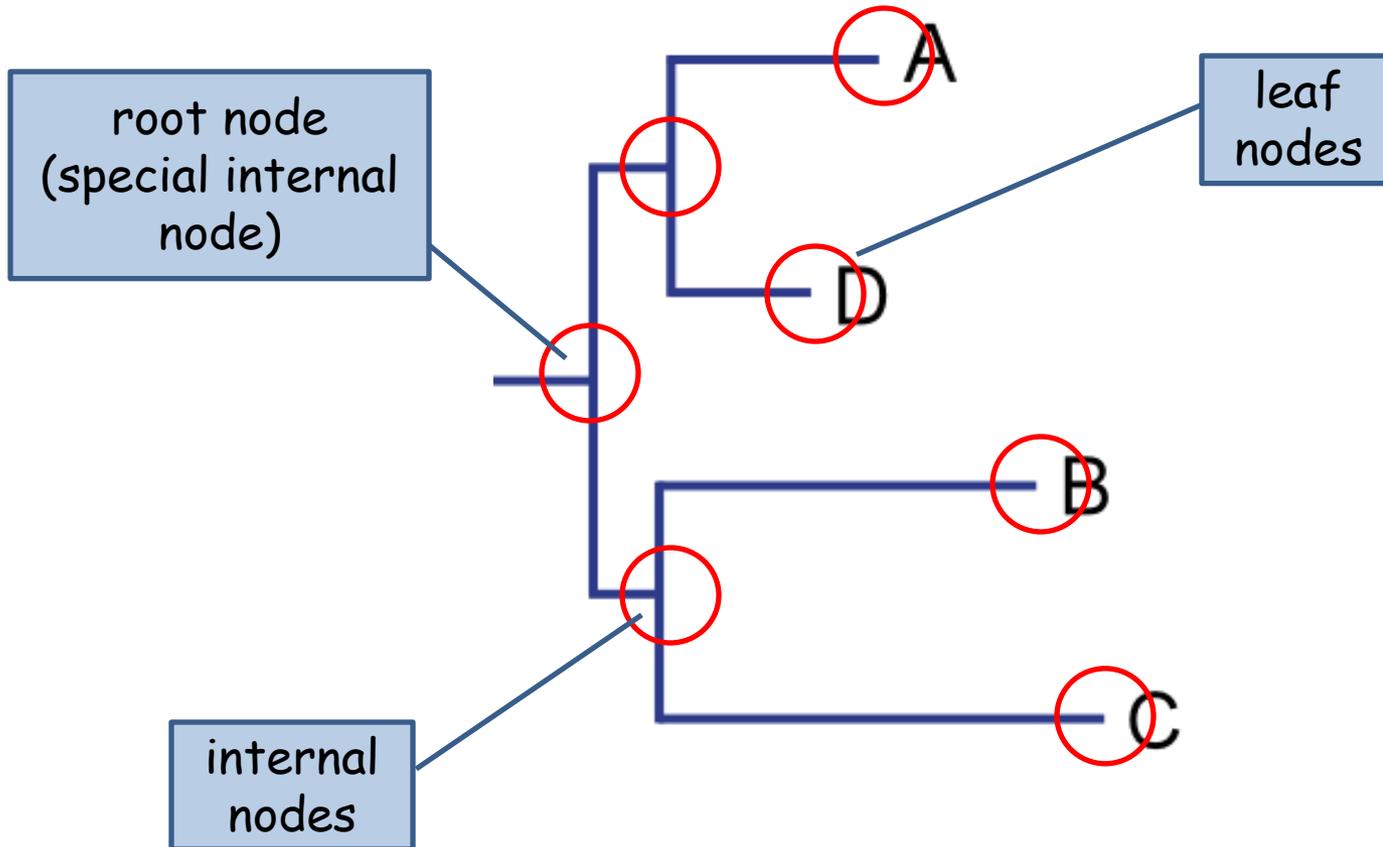
Some bioinformatic entities are easy to represent with standard Python types, e.g. :

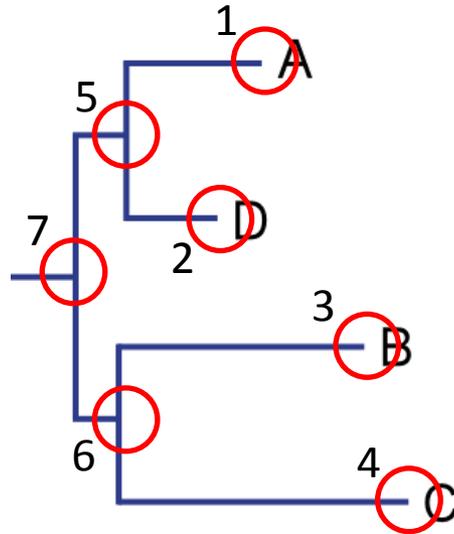
- Protein or DNA sequence
- Alignment score
- Sequence names paired with scores (or other things)

How would you represent a tree??



# Natural approach - represent tree nodes





tree nodes  
numbered for  
reference

What kinds of information should we associate with nodes?

- 1) A sequence name (for leaf nodes)
- 2) A distance to parent (except for the root)
- 3) Connections to other nodes