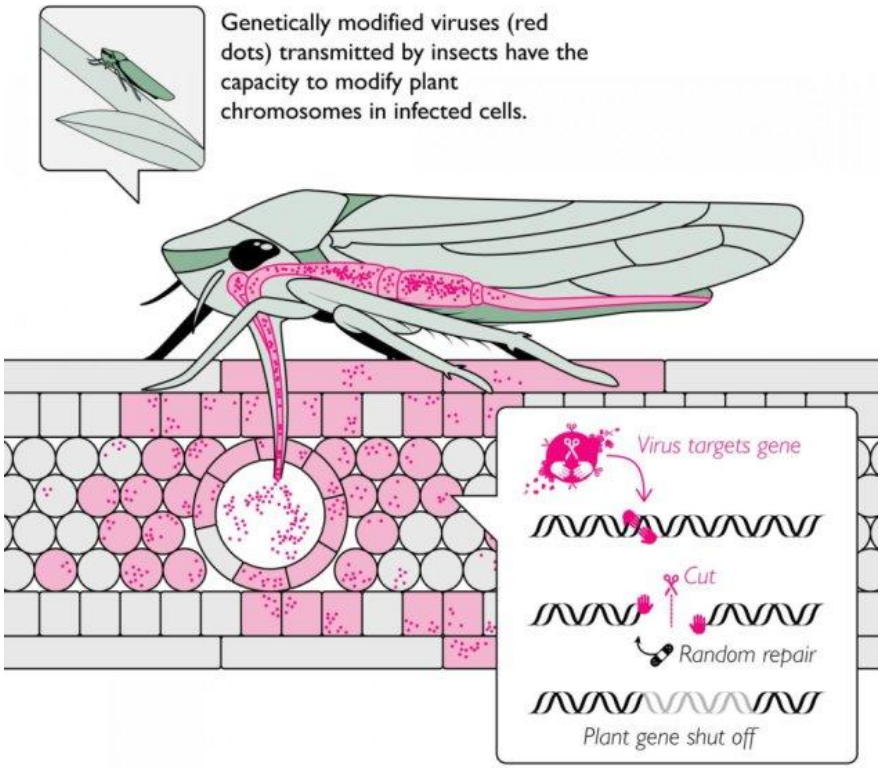
Phytobacteriology in the News

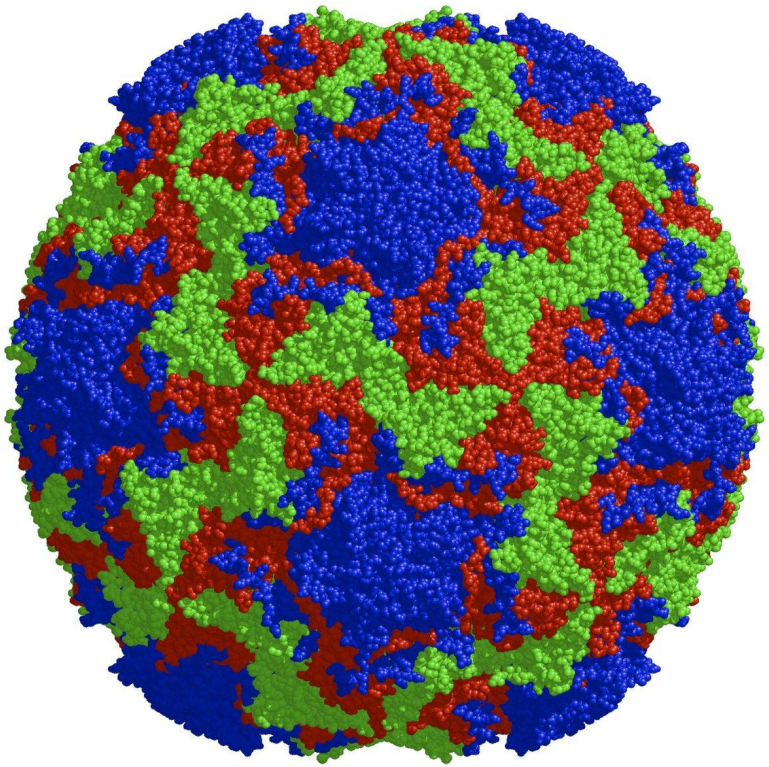
A step towards biological warfare with insects?

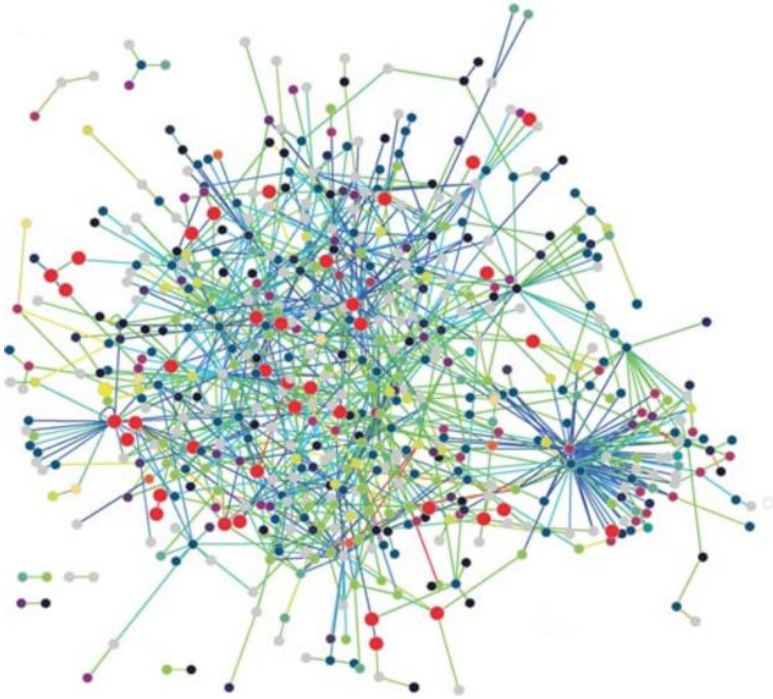


From the news story: "***A step towards biological warfare with insects?***" October 11, 2018. https://www.sciencedaily.com/releases/2018/10/181003102709.htm

Original scientific paper: R. G. Reeves, S. Voeneky, D. Caetano-Anollés, F. Beck, C. Boëte. **Agricultural research, or a new bioweapon system?** *Science*, 2018; 362 (6410): 35 DOI: 10.1126/science.aat7664

The programme called 'Insect Allies' intends for insects to be used for dispersing genetically modified viruses to agricultural plants in fields. These viruses would be engineered so they can alter the chromosomes of plants through 'genome editing'. This would allow for genetic modifications to be implemented quickly and at a large scale on crops that are already growing in fields, such as corn. In the journal *Science*, scientists from the Max Planck Institute for Evolutionary Biology in Plön and the Institut des Sciences de l'Evolution de Montpellier along with legal scholars from the University of Freiburg point out that this type of system could be more easily developed for use as a biological weapon than for the proposed agricultural purpos e.It is argued by the programs funders, that genome editing using synthetic viruses will open up unprecedented possibilities for changing the properties of crop plants already growing in fields. Plants could, for example, be genetically altered to nearly instantly become less susceptible to pests or droughts. Until now, genetic engineering of commercial seeds always occurred in laboratories. With farmers planting seeds, needing to anticipate what environmental conditions will likely arise during a growing season. This means that, in the case of an unexpected drought, only farmers who had already planted drought-resistant seeds would gain a benefit. However, the originators of this project claim that genetic engineering in fields would offer farmers the possibility to alter the genetic properties of their crops at any time. Use of this technology would represent a radical break with many existing farming practices, potentially jeopardizing their coexistence.





Guest lecturer: Dr. Richard Allen White III
Post-doc: Friesen lab

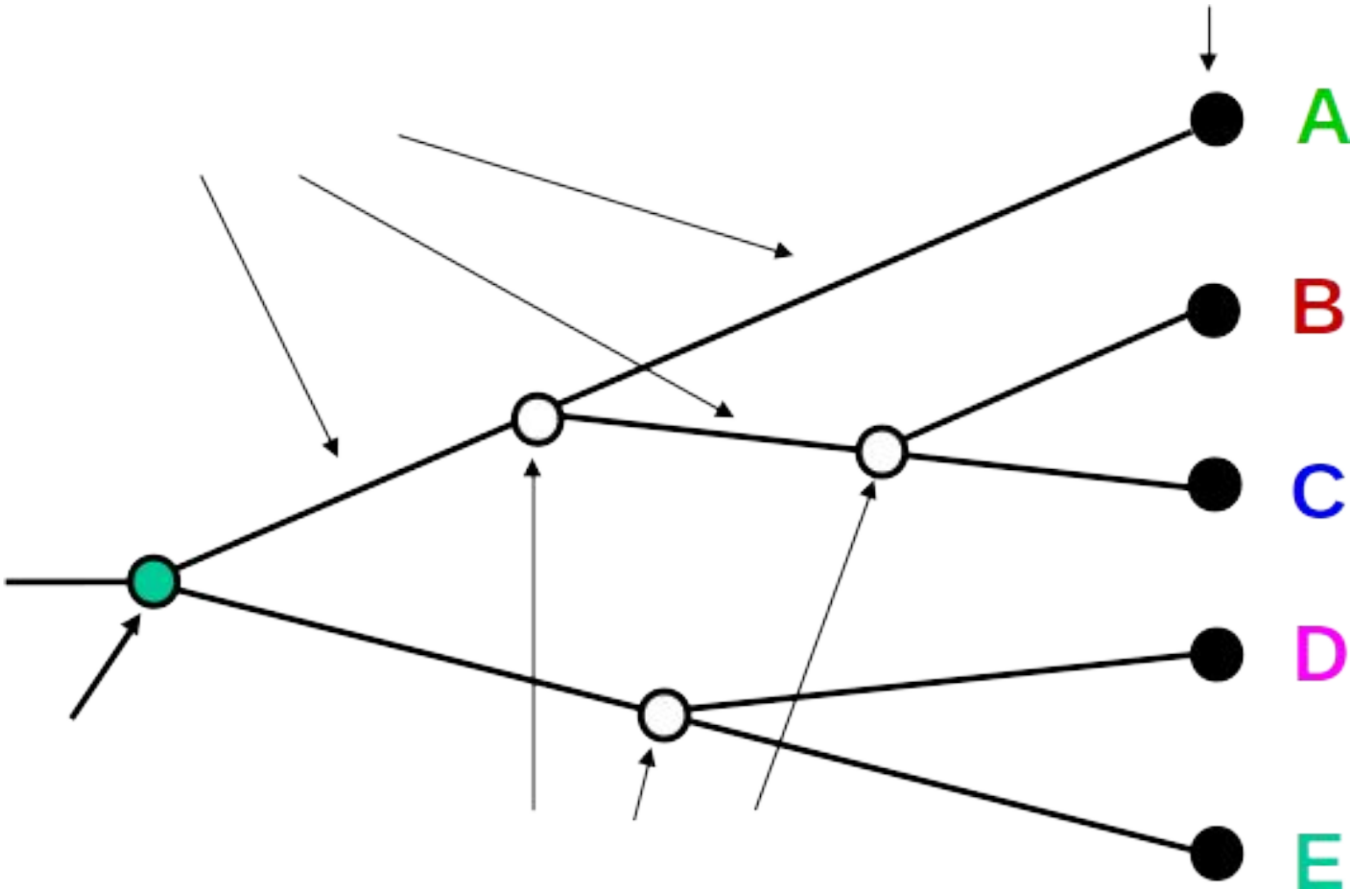
Announcements

- Phytobacteriology in the News? Anyone?
- News & Views due next ***Thursday***
- Thoughts on Sheng-Yang He lecture? Questions for Maren to email him?

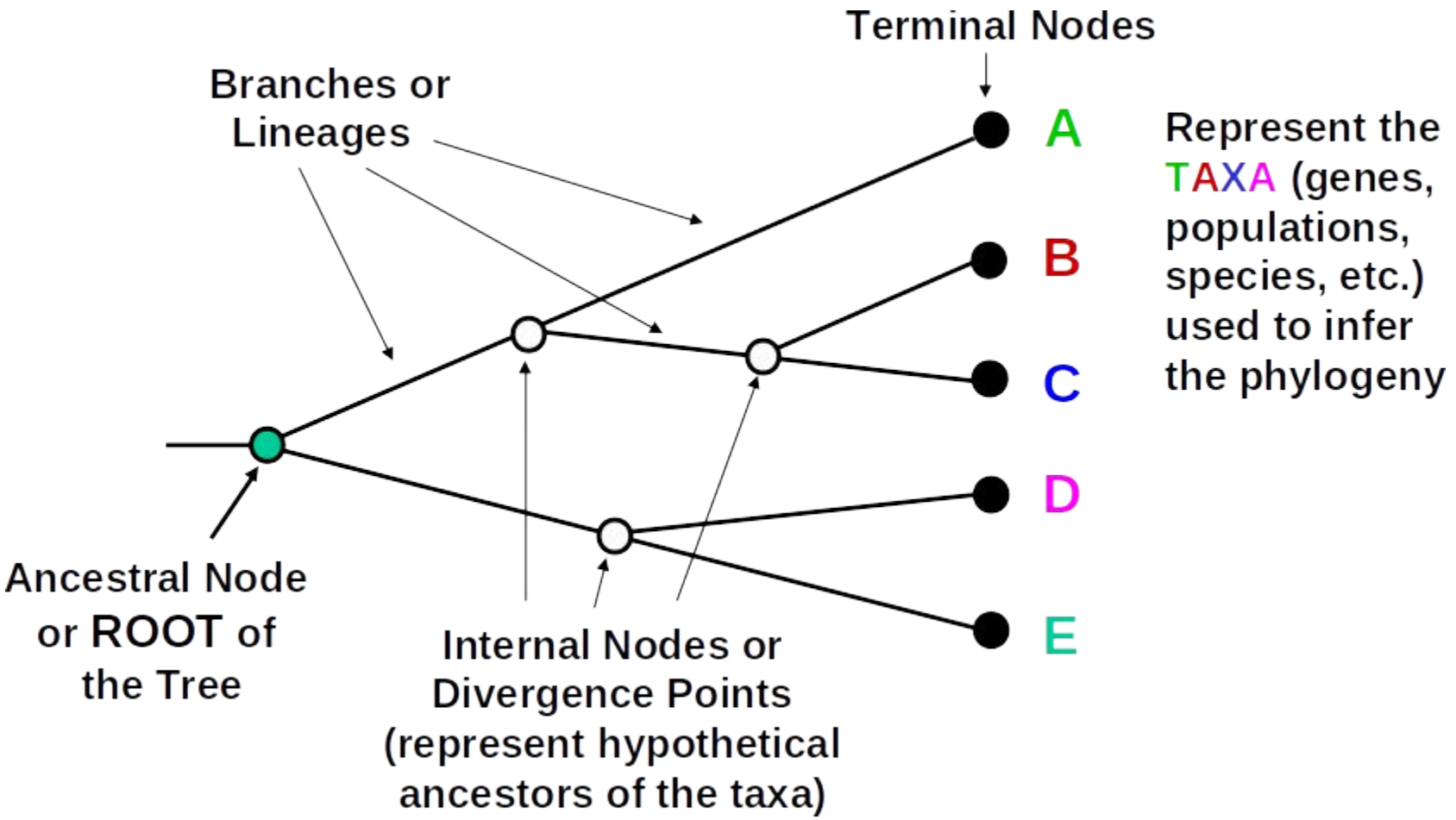
Learning Objectives

1. Tree terminology
2. Rooting trees
3. Evolutionary relationships
4. Tree building and methods
5. MAFFT via command line
6. Iqtree via command line

Terminology

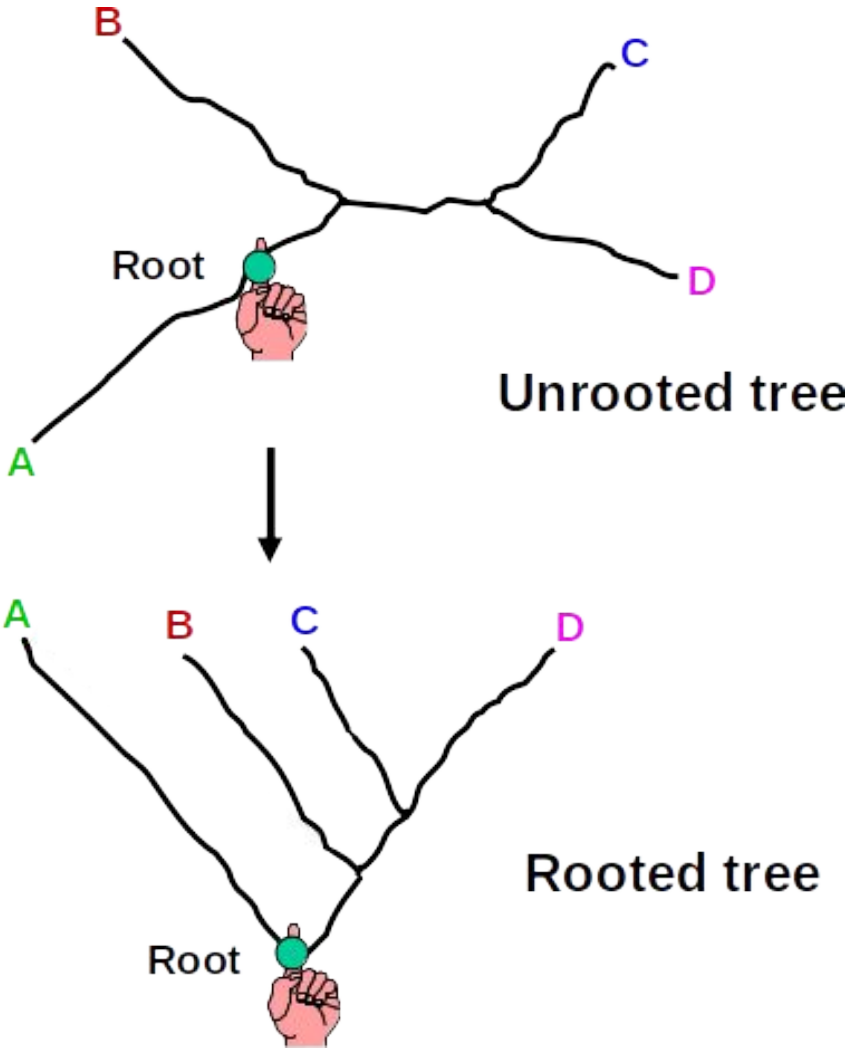


Terminology



Rooting trees

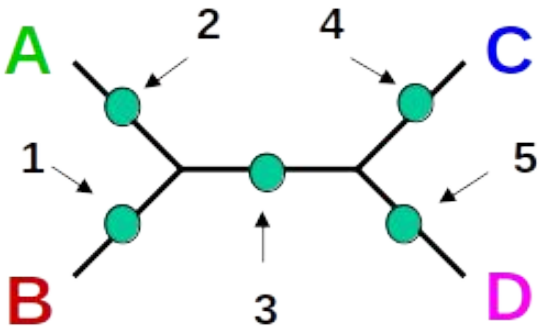
To root a tree mentally, imagine that the tree is made of string. Grab the string at the root ● and tug on it until the ends of the string (the taxa) fall opposite the root:



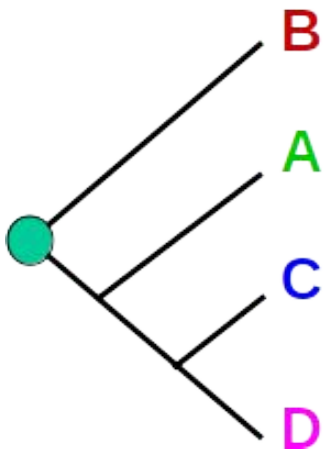
Note that in this rooted tree, taxon A is no more closely related to taxon B than it is to C or D.

Rooting trees

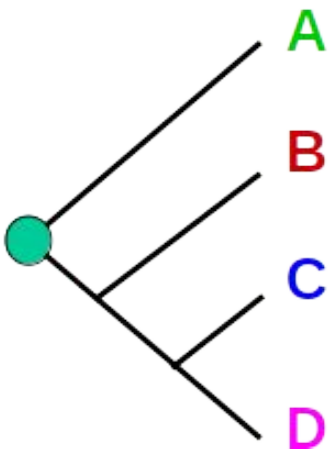
The unrooted tree 1:



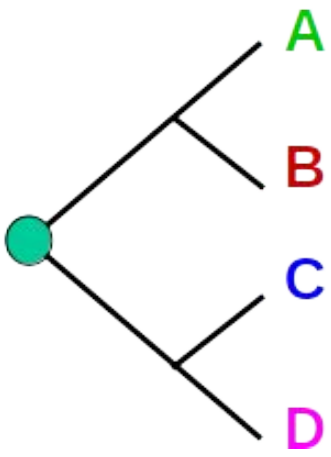
Rooted tree 1a



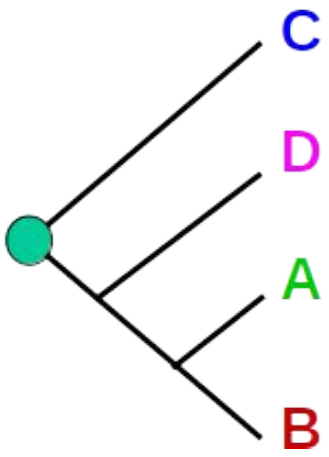
Rooted tree 1b



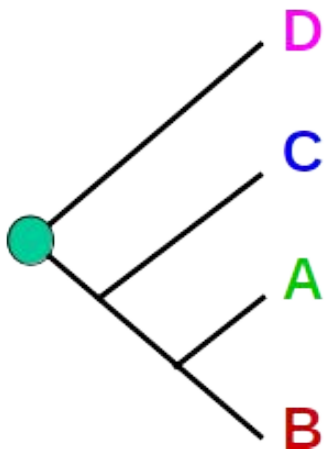
Rooted tree 1c



Rooted tree 1d

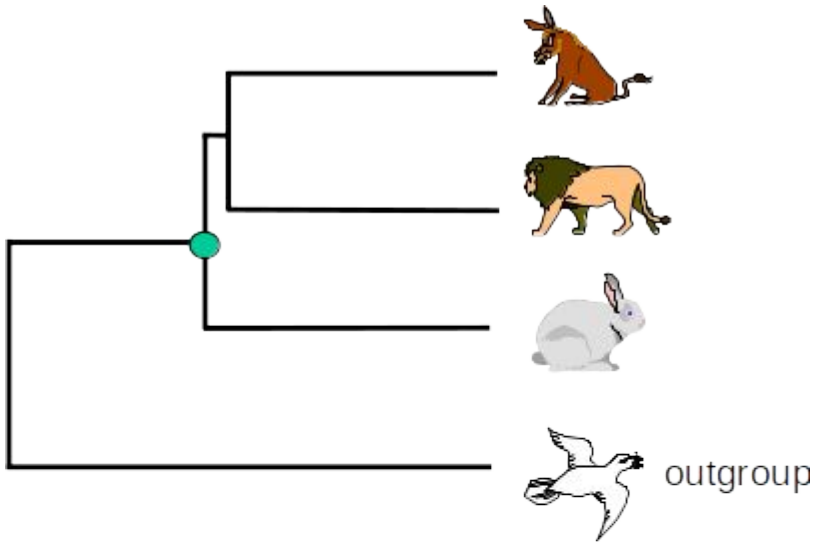


Rooted tree 1e

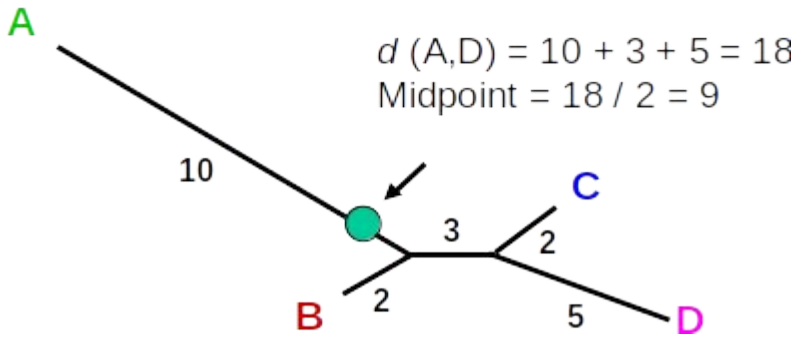


Rooting trees

By outgroup: Uses taxa (the "outgroup") that are known to fall outside of the group of interest (the "ingroup"). Requires some prior knowledge about the relationships among the taxa.

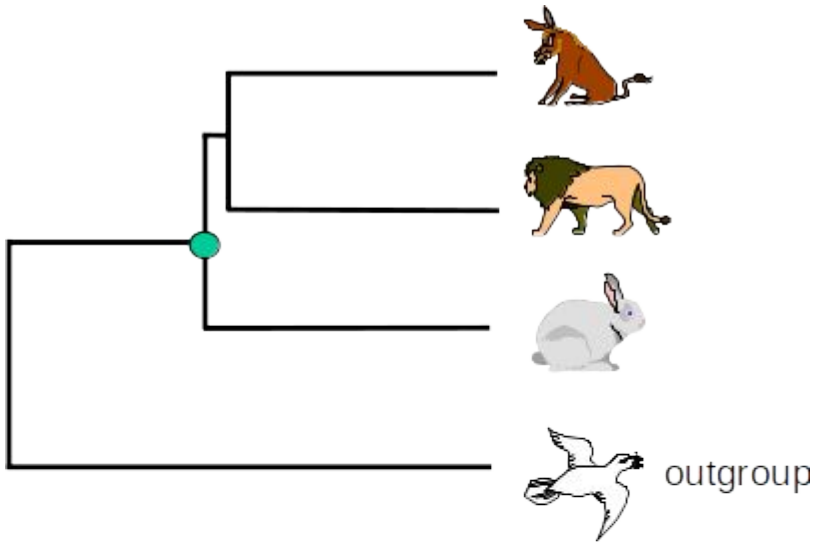


By midpoint or distance: Roots the tree at the midway point between the two most distant taxa in the tree, as determined by branch lengths. Assumes that the taxa are evolving in a clock-like manner. This assumption is built into some of the distance-based tree building methods.

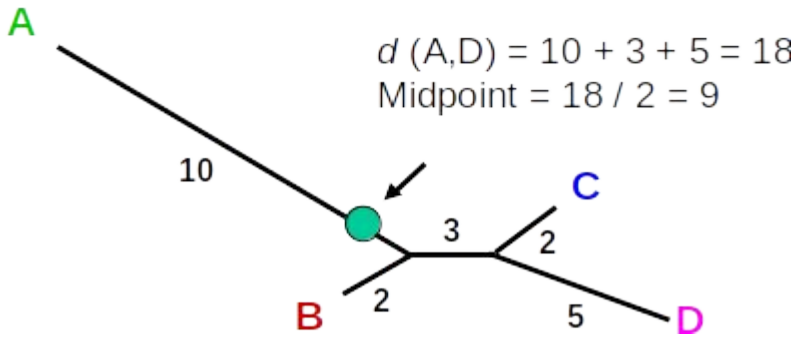


Rooting trees

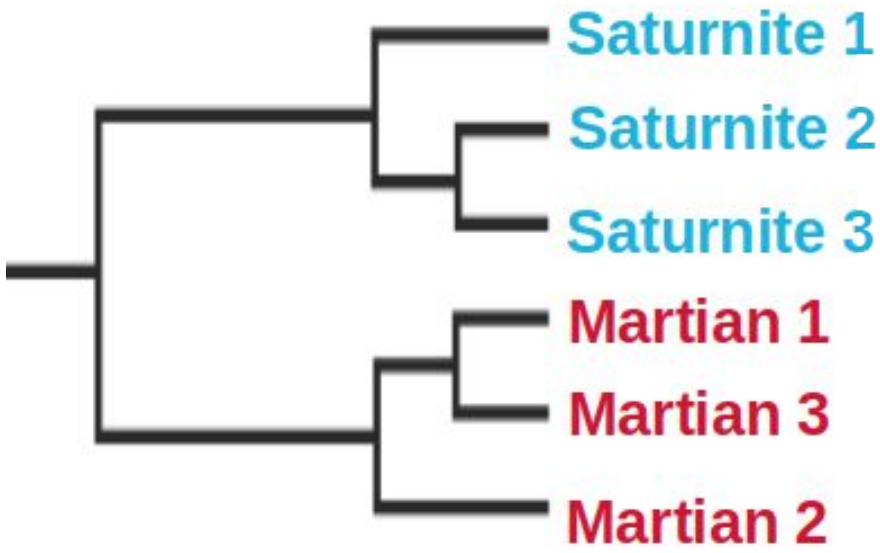
By outgroup: Uses taxa (the "outgroup") that are known to fall outside of the group of interest (the "ingroup"). Requires some prior knowledge about the relationships among the taxa.

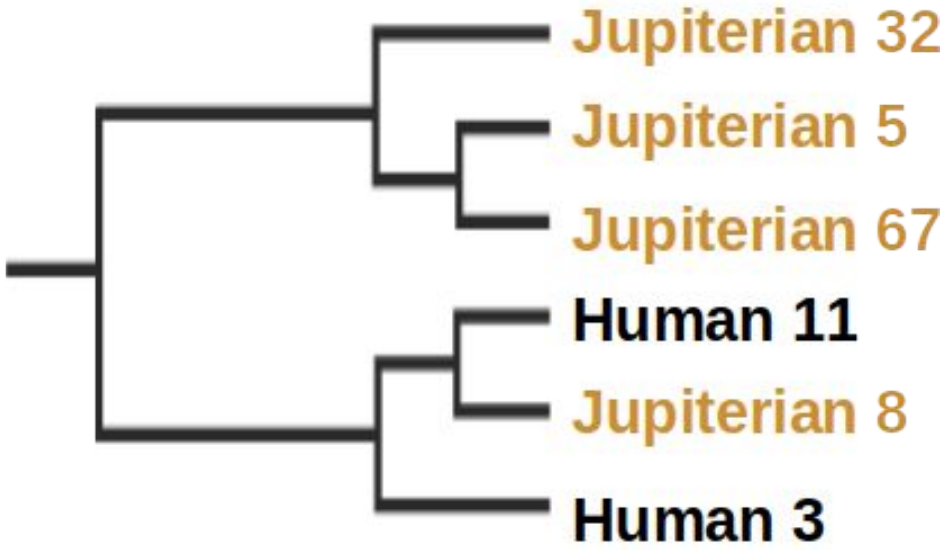


By midpoint or distance: Roots the tree at the midway point between the two most distant taxa in the tree, as determined by branch lengths. Assumes that the taxa are evolving in a clock-like manner. This assumption is built into some of the distance-based tree building methods.



Evolutionary relationships



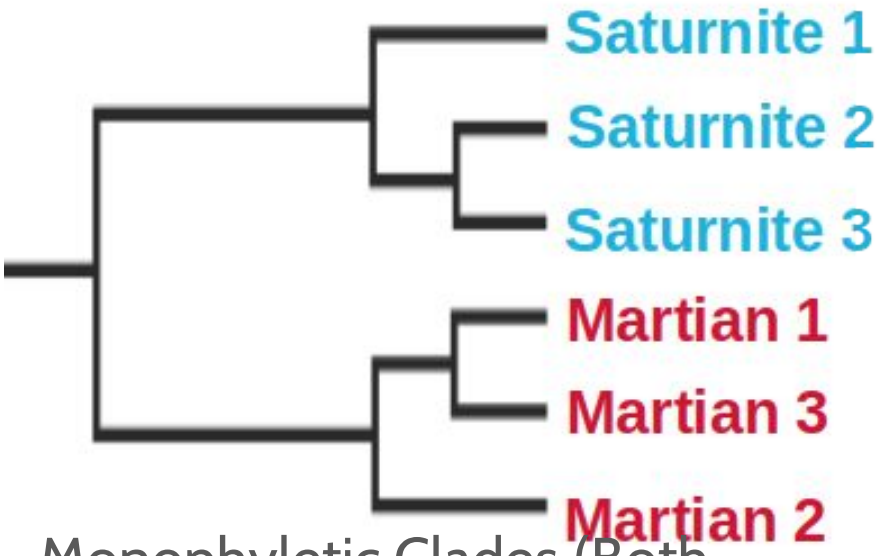


Monophyletic?

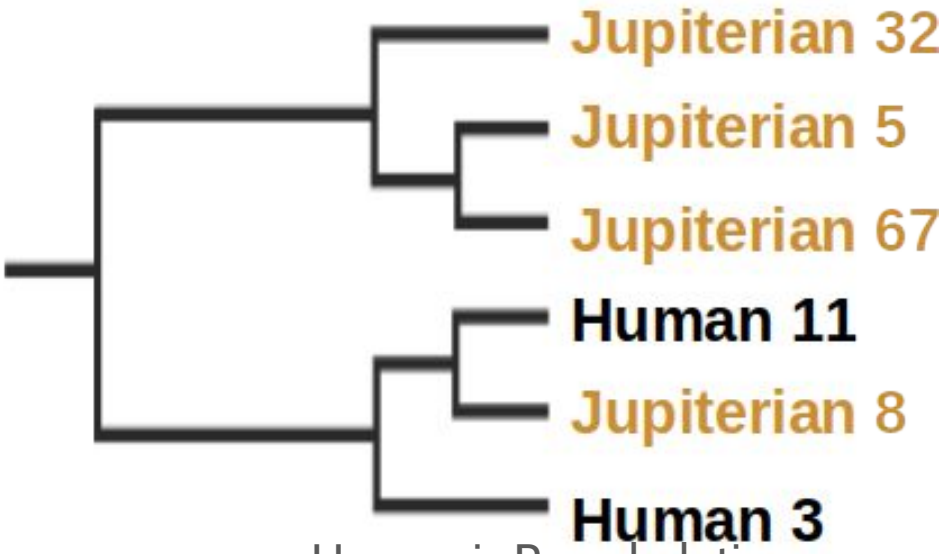
Paraphyletic?

Polyphyletic?

Evolutionary relationships



Monophyletic Clades (Both Saturnites & Martians)



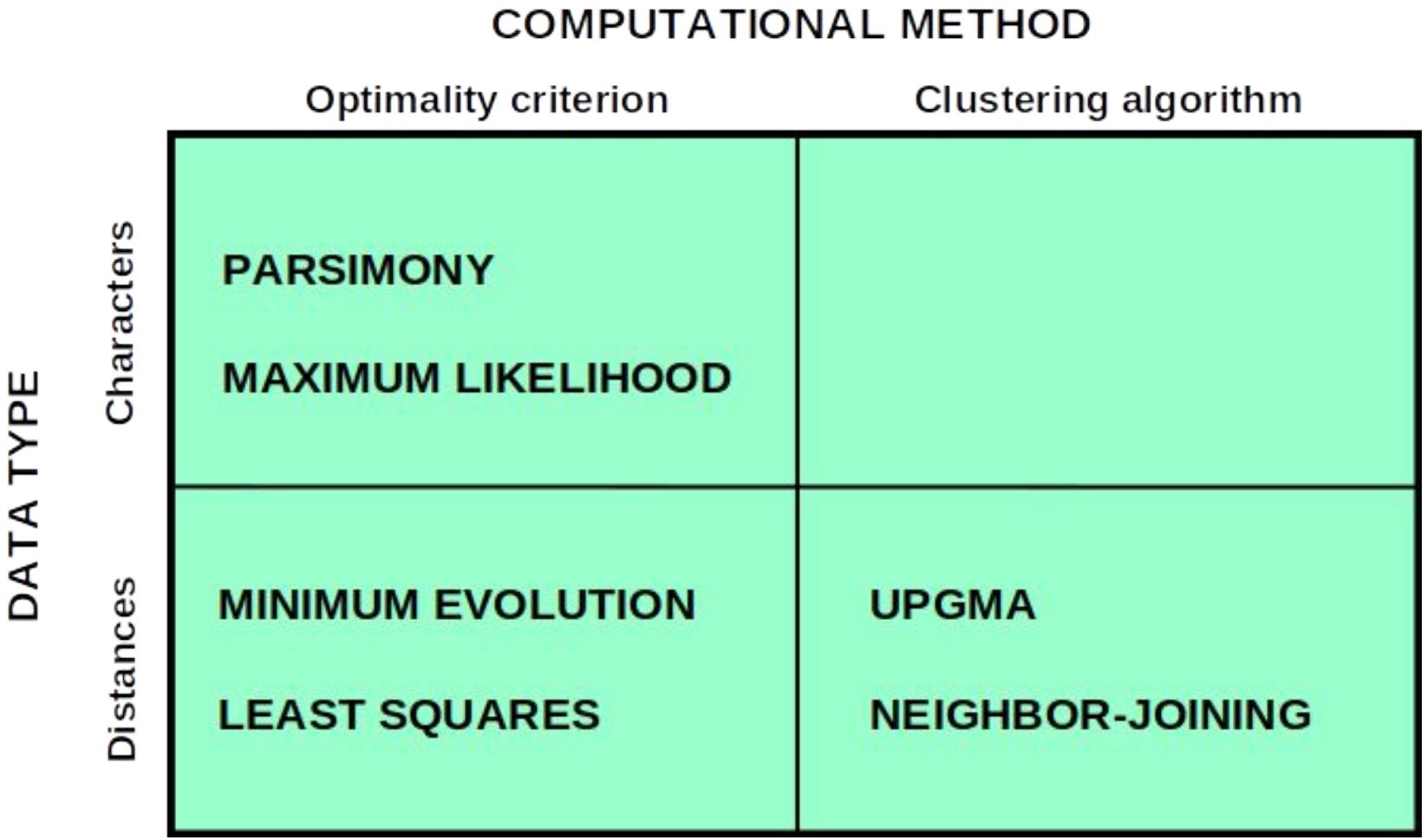
Human is Paraphyletic
Jupiterian is Polyphyletic

Monophyletic CLADE: (of a group of organisms) descended from a common evolutionary ancestor or ancestral group, especially one not shared with any other group.

Paraphyletic CLADE: (of a group of organisms) descended from a common evolutionary ancestor or ancestral group, but not including all the descendant groups.

Polyphyletic CLADE: (of a group of organisms) derived from more than one common evolutionary ancestor or ancestral group and therefore not suitable for placing in the same taxon

Tree building



Tree building

Character-based methods: Use the aligned characters, such as DNA or protein sequences, directly during tree inference.

Taxa	Characters
Species A	ATGGCTATTCTTATAGTACG
Species B	ATCGCTAGTCTTATATTACA
Species C	TTCACTAGACCTGTGGTCCA
Species D	TTGACCAGACCTGTGGTCCG
Species E	TTGACCAGTTCTCTAGTTCG

Distance-based methods: Transform the sequence data into pairwise distances (dissimilarities), and then use the matrix during tree building.

	A	B	C	D	E
Species A	----	0.20	0.50	0.45	0.40
Species B	0.23	----	0.40	0.55	0.50
Species C	0.87	0.59	----	0.15	0.40
Species D	0.73	1.12	0.17	----	0.25
Species E	0.59	0.89	0.61	0.31	----

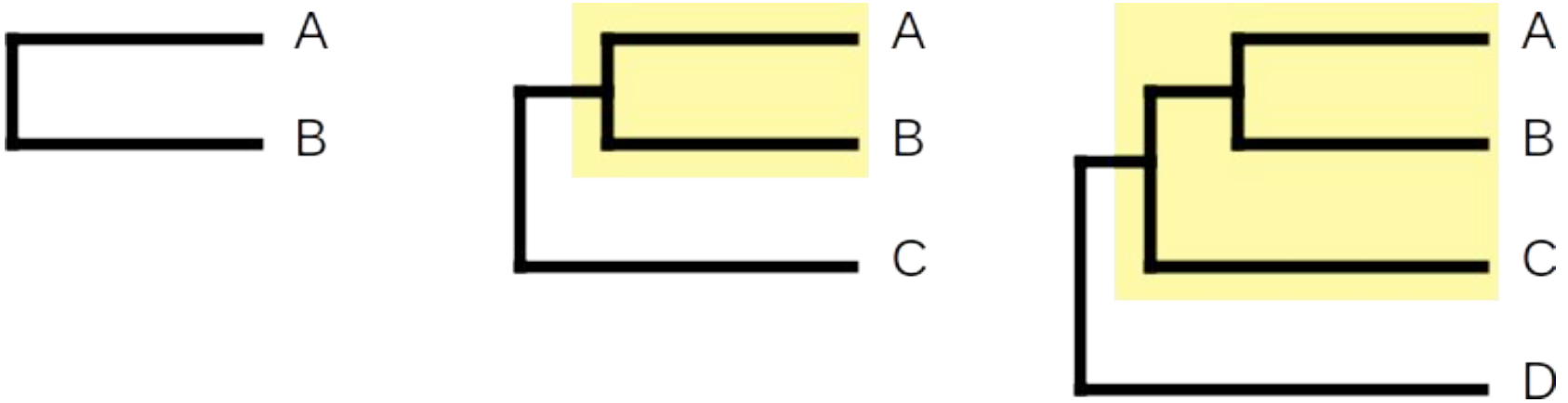
Methods for Tree building

1. Distance
2. Maximum parsimony
3. Maximum likelihood
4. Bayesian

Methods for Tree building

1. Distance

Using a sequence alignment, pairwise distances are calculated Creates a distance matrix A phylogenetic tree is calculated with clustering algorithms, using the distance matrix. Examples of clustering algorithms include the Unweighted Pair Group Method using Arithmetic averages (UPGMA) and Neighbor Joining clustering.

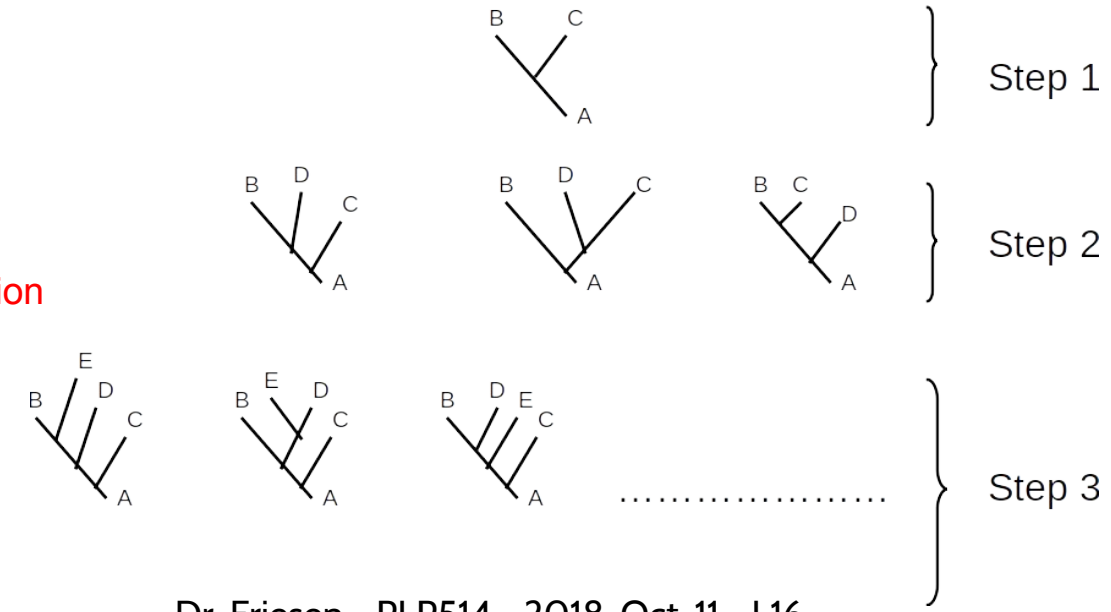


Methods for Tree building

2. Maximum Parsimony

a) All possible trees are determined for each position of the sequence alignment
b) Each tree is given a score based on the number of evolutionary step needed to produce said tree
c) The most parsimonious tree is the one that has the fewest evolutionary changes for all sequences to be derived from a common ancestor
d) Usually several equally parsimonious trees result from a single run.

exhaustive stepwise addition



Methods for Tree building

3. Maximum Likelihood

a) Creates all possible trees like Maximum Parsimony method but instead of retaining trees with shortest evolutionary steps.
b) Employs a model of evolution whereby different rates of transition/transversion ratio can be used. Each tree generated is calculated for the probability that it reflects each position of the sequence data. Calculation is repeated for all nucleotide sites
c) It is a more realistic tree estimation because it does not assume equal transition-transversion ratio for all branches.
d) Advantages:
    i) Highly accurate because considerable biological realism is introduced through the substitutional model. This allows various forms of homoplasy to be corrected for.
    ii) Phylogenetic estimation within the likelihood framework provides a robust statistical context in which to evaluate specific hypotheses.
    iii) A single tree is produced that is generally precise.
e) Disadvantages
    i) The complexity of the estimation process means that it is slow and computationally demanding.
    ii) The hill-climbing algorithm is susceptible to local optima and so does not guarantee to return the most optimal solution.

Methods for Tree building

Distance	Maximum parsimony	Maximum likelihood
Uses only pairwise distances	Uses only shared derived characters	Uses all data
Minimizes distance between nearest neighbors	Minimizes total distance	Maximizes tree likelihood given specific parameter values
Very fast	Slow	***Very*** slow
Easily trapped in local optima	Assumptions fail when evolution is rapid	Highly dependent on assumed evolution model
Good for generating tentative tree, or choosing among multiple trees	Good option when tractable (<30 taxa, homoplasy rare)	Best option. Usually for very small data sets and for testing trees built using other methods

Methods for Tree building

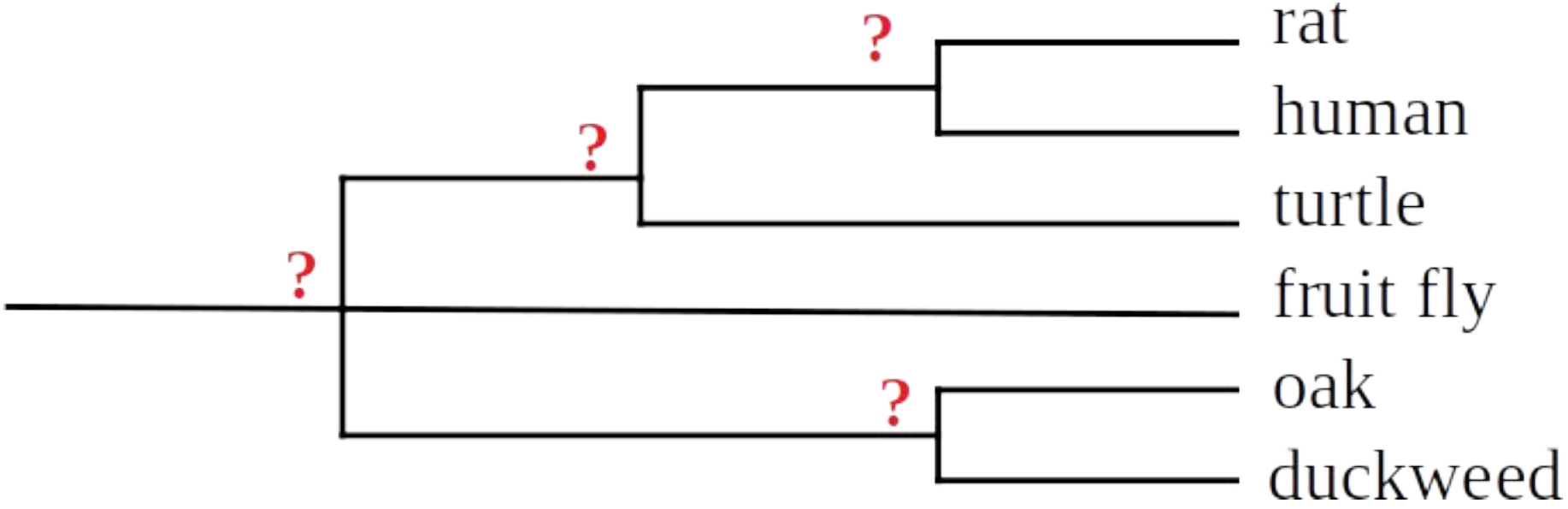
4. Bayesian

a) Based on the notion of posterior probabilities: probabilities that are estimated, based on some model (prior expectations), after learning something about the data. Uses an MCMC process to search through tree-space. Selects the tree-topology with the highest probability, given the data.

b) Advantages
   i) Posterior probabilities describe the absolute probability of particular nodes and branch lengths; these can be overestimated.
   ii) Intuitive Potential for any complex model.
   iii) Provides both parameter estimates (i.e., tree) and their probabilities in a single analysis. Many different hypotheses can be evaluated in a single analysis. The MCMC algorithm makes integrating over all parameter values fast and accurate; MCMCs are able to break out of local optima.

c) Disadvantages
   i) An evolutionary model must be specified a priori, in form of prior probabilities ('priors'). Is there sufficient knowledge of these probabilities?
   ii) The MCMC must be run long enough for variation in the parameter estimates to smooth out or reach 'convergence'. The time required is never certain.

Bootstrapping



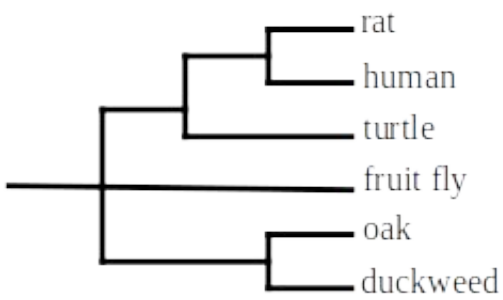
How confident are we?

Bootstrapping

Bootstrapping relies on random sampling with replacement to estimate confidence. **Bootstrapping** allows assigning measures of accuracy (defined in terms of bias, variance, confidence intervals, prediction error or some other such measure) to sample estimates.

Sample

```
           0123456789
rat        GAGGCTTATC
human      GTGGCTTATC
turtle     GTGCCCTATG
fruitfly   CTCGCCTTTG
oak        ATCGCTCTTG
duckweed   ATCCCTCCGG
```



Inferred tree

Pseudo sample 1

```
           001122234556667
rat        GGAAGGGGCTTTTTA
human      GGTTGGGGCTTTTTA
turtle     GGTTGGGCCCCTTTA
fruitfly   CCTTCCCGCCCTTTT
oak        AATTCCCGCTTCCCT
duckweed   AATTCCCCCTTCCCC
```

Pseudo sample 2

```
           445556777888899
rat        CCTTTTAAATTTTCC
human      CCTTTTAAATTTTCC
turtle     CCCCCTAAATTTTGG
fruitfly   CCCCCTTTTTTTTGG
oak        CCTTTCTTTTTTTGG
duckweed   CCTTTCCCCGGGGGG
```

Many more replicates (between 100 - 1000)

Run MAFFT in command line

Options -> for accuracy methods

*L-INS-i (probably most accurate; recommended for <200 sequences; iterative refinement method incorporating local pairwise alignment information):
mafft --localpair --maxiterate *1000 input* [> *output*]
linsi *input* [> *output*]

*G-INS-i (suitable for sequences of similar lengths; recommended for <200 sequences; iterative refinement method incorporating global pairwise alignment information):
mafft --globalpair --maxiterate *1000 input* [> *output*]
ginsi *input* [> *output*]

*E-INS-i (suitable for sequences containing large unalignable regions; recommended for <200 sequences):
mafft --ep *0* **--genafpair --maxiterate** *1000 input* [> *output*]
einsi *input* [> *output*]

Run MAFFT in command line

Options -> for speed methods

*FFT-NS-i (iterative refinement method; two cycles only):
mafft --retree *2* **--maxiterate** *2 input* [> *output*]
fftnsi *input* [> *output*]

*FFT-NS-i (iterative refinement method; max. 1000 iterations):
mafft --retree *2* **--maxiterate** *1000 input* [> *output*]

*FFT-NS-2 (fast; progressive method):
mafft --retree *2* **--maxiterate** *0 input* [> *output*]
fftns *input* [> *output*]

*FFT-NS-1 (very fast; recommended for >2000 sequences; progressive method with a rough guide tree):
mafft --retree *1* **--maxiterate** *0 input* [> *output*]

*NW-NS-i (iterative refinement method without FFT approximation; two cycles only):
mafft --retree *2* **--maxiterate** *2* **--nofft** *input* [> *output*]
nwnsi *input* [> *output*]

*NW-NS-2 (fast; progressive method without the FFT approximation):
mafft --retree *2* **--maxiterate** *0* **--nofft** *input* [> *output*]
nwns *input* [> *output*]

*NW-NS-PartTree-1 (recommended for ~10,000 to ~50,000 sequences; progressive method with the PartTree algorithm):
mafft --retree *1* **--maxiterate** *0* **--nofft --parttree** *input* [> *output*]

Run MAFFT in command line

./mafft --localpair --maxiterate 1000 16S_nucleotide.fasta >16S_nucleotide.clust

Try with protein and the --globalpair

View alignment

-> http://wasabiapp.org/

-> https://www.ebi.ac.uk/Tools/msa/mview/

-> MEGA (windows)

-> Seaview (linux)

Run Iq-tree2 in command line

./iqtree -s 16S_nucleotide.clust -st DNA -m TEST -bb 1000 -alrt 1000

View tree

-> MEGA (windows)

-> Seaview (linux)