TUTORIAL 2: Tree Inference

Building Phylogenetic Trees using RAxML and FastTree algorithms

Assignment

- Build trees for the 16S rRNA sequence using RaxML and FastTree algorithms
- Explain how underlying algorithms work
- Comment on differences in trees based off the different algorithms

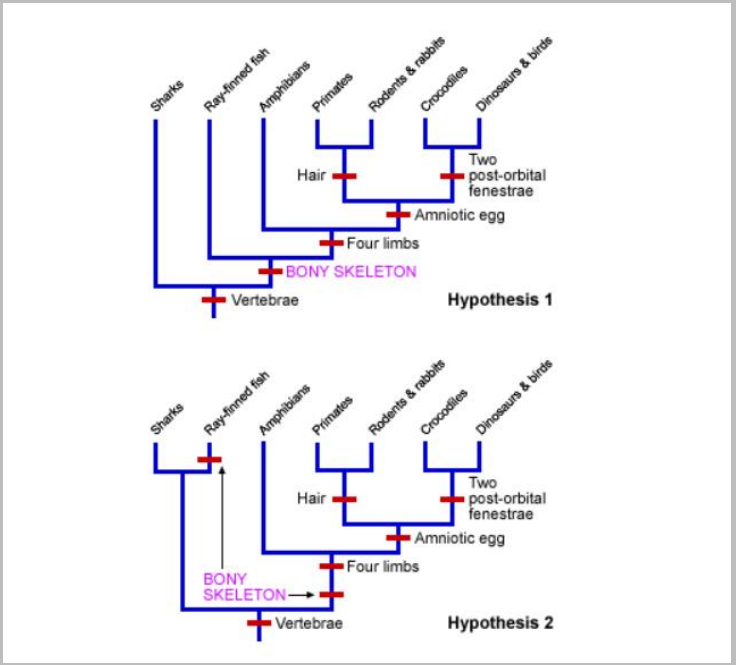
What are Trees?

- Trees show how groups of organisms, nucleic acids, or protein sequences are related evolutionarily
- Show how these groups came about through evolution
- Allows scientists to follow evolutionary changes that have occurred over time

Types of Tree Algorithms

- Maximum Likelihood (ML) uses an evolutionary model and tree topology finds the tree which has the highest likelihood of giving the observed data
- ML scores represent the probability of the observed sequence data given a specific phylogenetic tree and evolutionary model
- Distance Methods calculate evolutionary distances and build trees based off of these distances. Are also much less computationally intensive
- Maximum Parsimony (MP) makes tree that has minimum number of changes to explain observed data

Parsimony



https://evolution.berkeley.edu/phylogenetic-systematics/reconstructing-trees-cladistics/reconstructing-trees-parsimony/

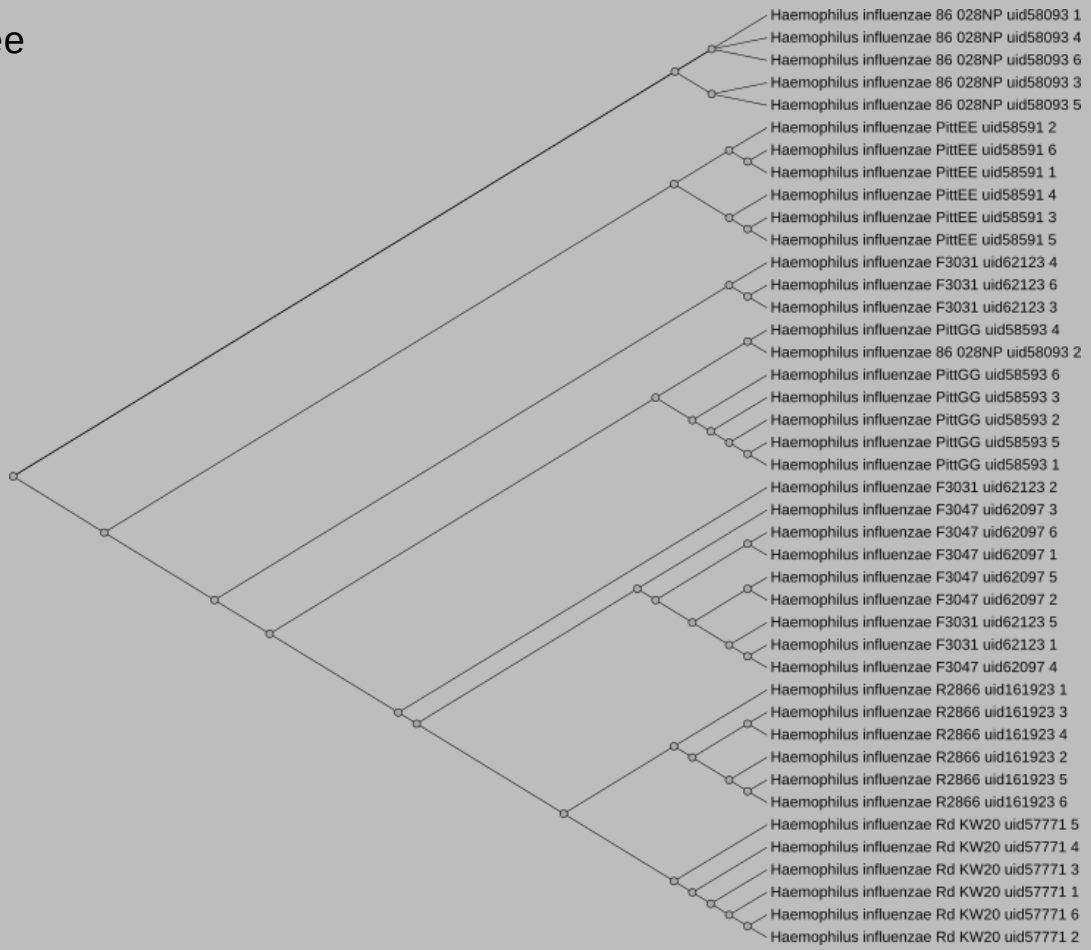
Bootstrapping

- Bootstrapping is like taking many different snapshots of your data by randomly selecting observations from it.
- Each bootstrap sample is the same size as the original dataset, but it's created by randomly selecting sites from the original dataset
- Some observations may be selected more than once, while others may not be selected at all
- Reconstruct a phylogenetic tree for each replicate, and then evaluate the frequency that branches appear
- The support value associated with each branch indicates the proportion of bootstrap replicates that support that specific branch. Higher support values indicate greater confidence in the presence of that branch in the true evolutionary tree.

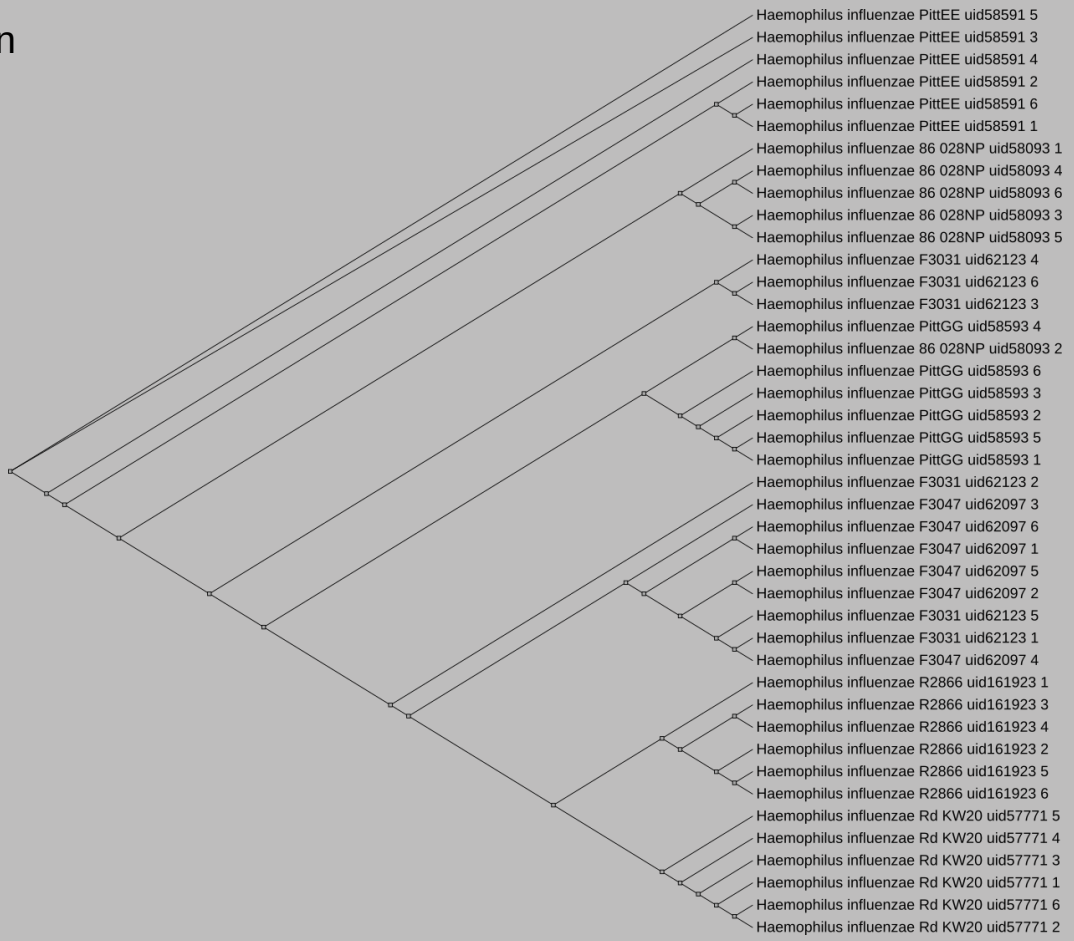
RAxML

- RAxML (**R**andomized **Ax**elerated **M**aximum **L**ikelihood) is an ML algorithm
- Uses randomly generated trees and parsimony based starting trees
- Uses variety of heuristic search algorithms to explore possible tree topologies and parameter values - Subtree prune and regraft (SPR) and Nearest-Neighbor Interchange (NNI)
- Iteratively adjusts the tree topology and branch lengths to improve the likelihood score
- RAxML can do bootstrapping to assess the confidence levels of each branch
- Can calculate Robinson-Foulds (RF) distances between tree iterations
- The current leading method for large-scale ML estimation
- Can require weeks for datasets with thousands of molecular sequences

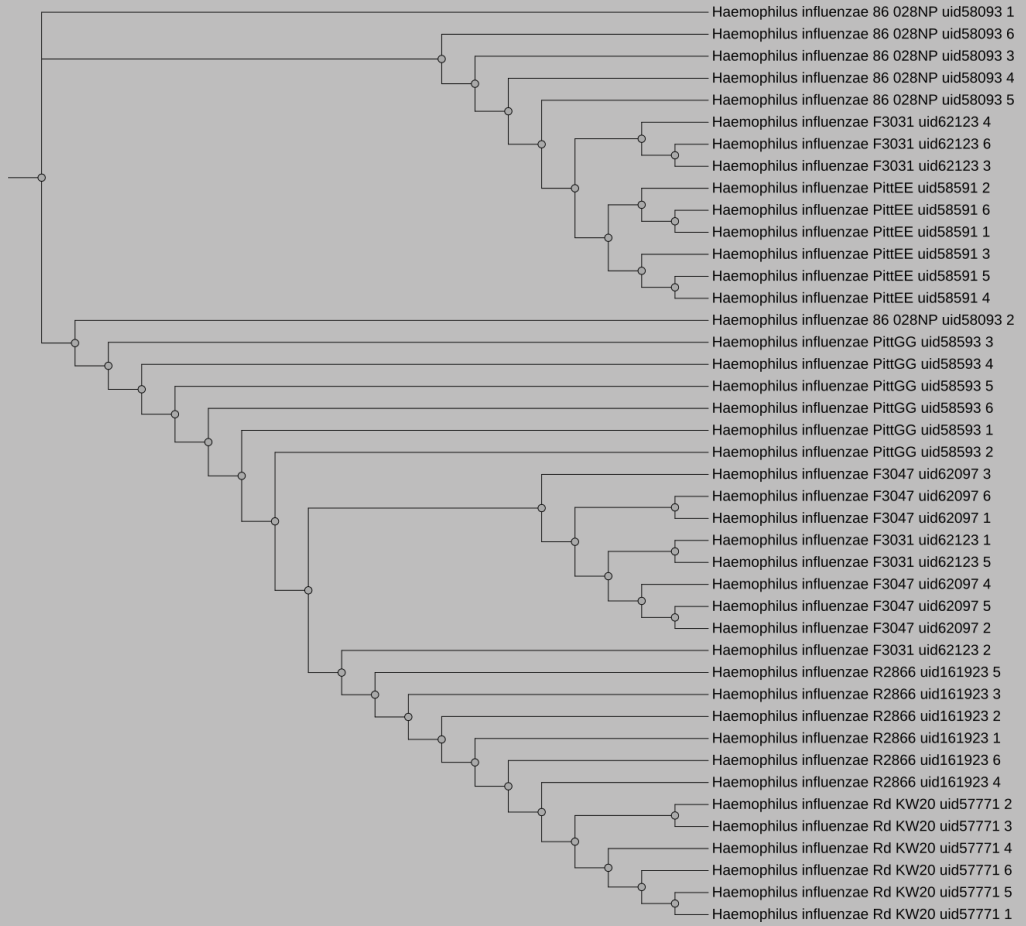
RAxML – Best Tree



RAxML – Bipartition



RAxML – Bootstrap

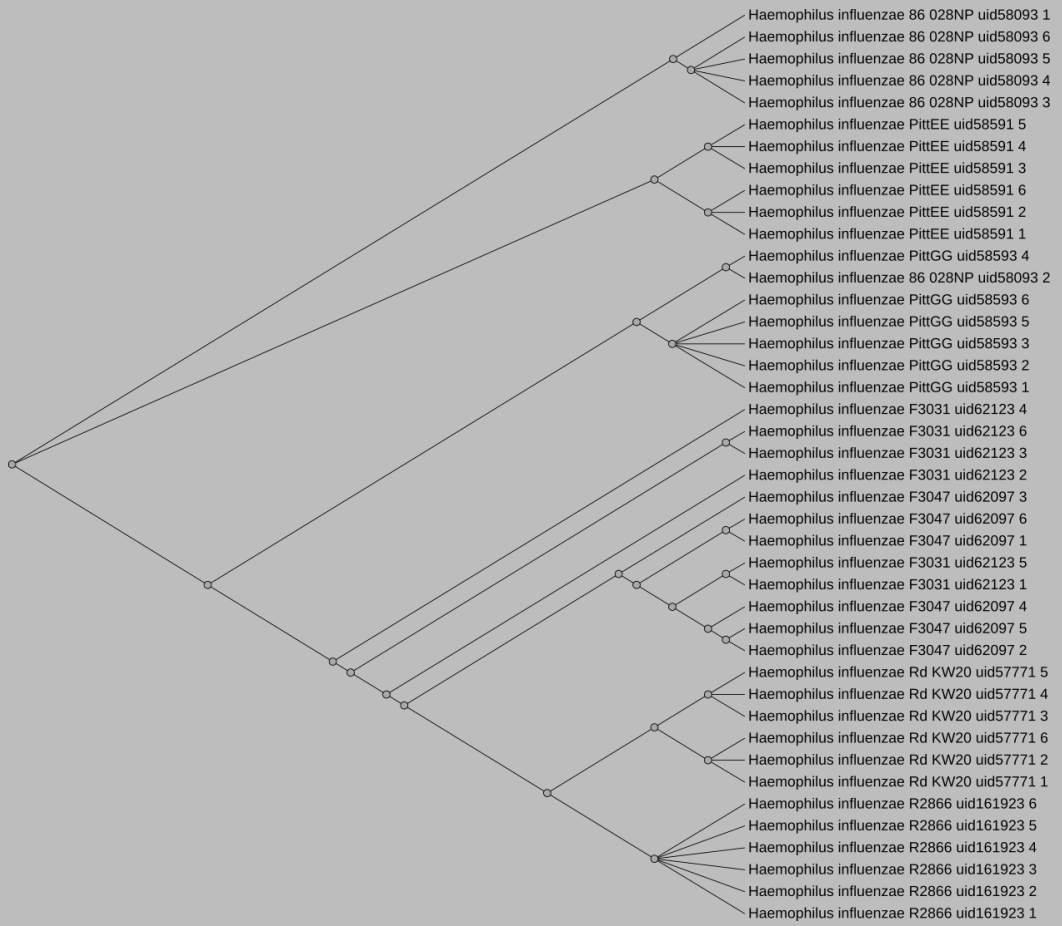


FastTree

- Algorithm used to construct phylogenetic trees from alignments of nucleotide or protein sequences
- Specifically designed to efficiently build phylogenetic trees from large datasets
- Can handle alignments with up to a million sequences
- Jukes-Cantor or generalized time-reversible (GTR) models of nucleotide evolution
- Jones-Taylor-Thornton (JJT), Whelan & Goldman (WAG), or Le and Gascuel (LG) models of amino acid evolutions
- Nearest Neighbor Interchange (NNI)
- A limitation of FastTree's neighbor joining phase is that it does not correct the distance for multiple for multiple substitutions, but it gets corrected in later steps

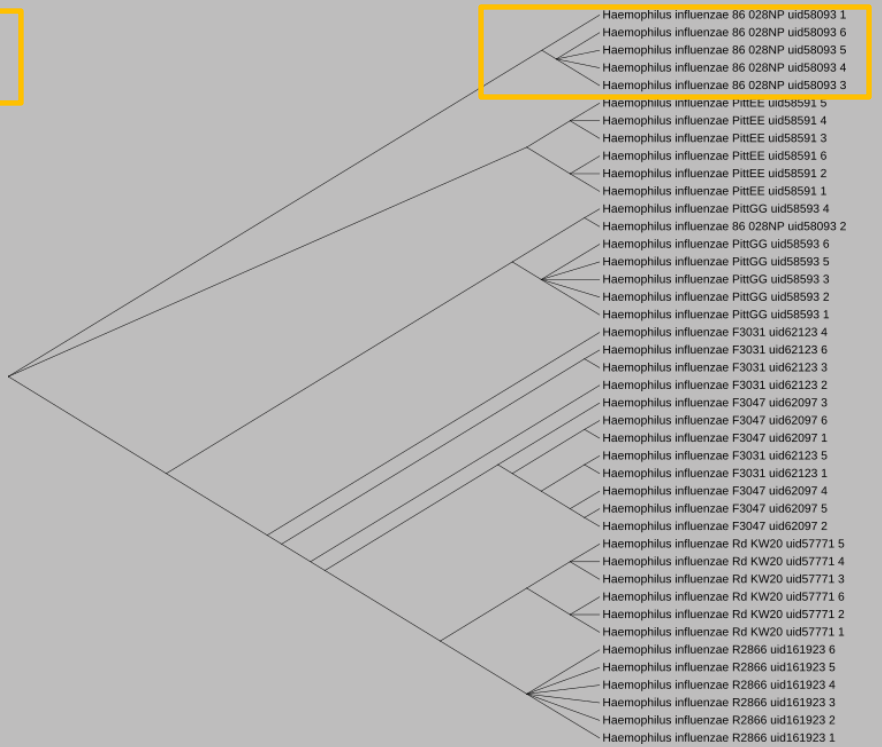
FastTree - Xsede

Rooted Tree
External Nodes: 42
Internal Nodes: 26



Comparing RAxML and FastTree

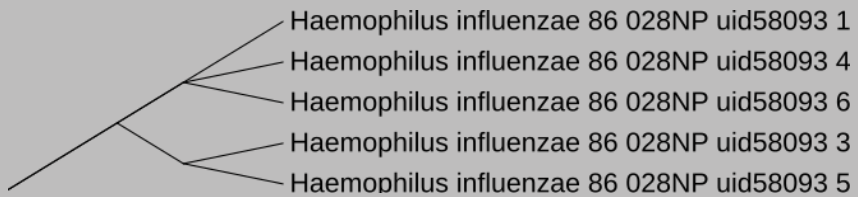
RAxML – Best Tree



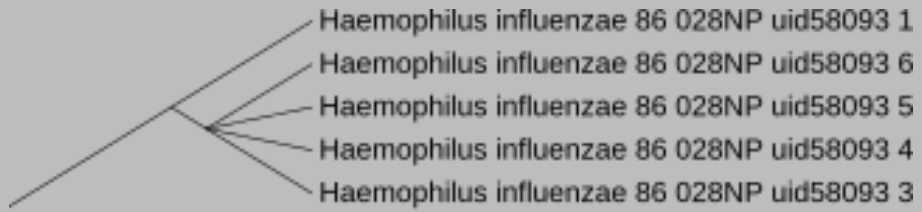
FastTree - Xsede

Comparing RAxML and FastTree

RAxML – Best Tree



FastTree - Xsede



Comparing RAxML and FastTree

- RAxML generally considered more accurate and FastTree considered faster
- "We find that when RAxML and FastTree are constrained to the same running time, FastTree produces topologically much more accurate trees in almost all cases. We also find that when RAxML is allowed to run to completion, it provides an advantage over FastTree in terms of the ML score, but does not produce substantially more accurate tree topologies."

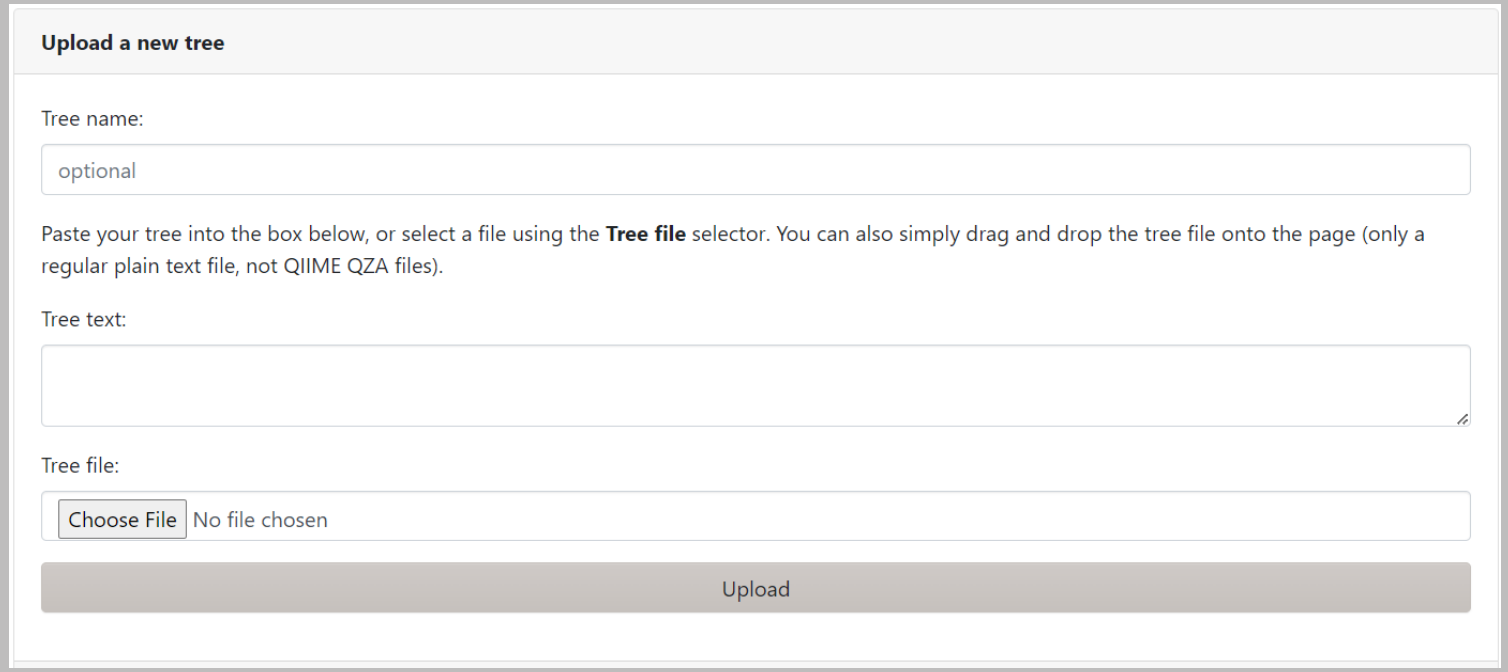
Build a Tree using iTOL

☐	PROCESS_OUTPUT	STDOUT	104.59 KB	View	Download
☐		STDERR	189.00 bytes	View	Download
☐	all_outputfiles	stderr.txt	189.00 bytes	View	Download
☐		RAxML_bestTree.result	3.59 KB	View	Download
☐		scheduler.conf	114.00 bytes	View	Download
☐		stdout.txt	104.59 KB	View	Download
☐		start.txt	40.00 bytes	View	Download
☐		RAxML_bootstrap.result	1.55 MB	View	Download
☐		done.txt	49.00 bytes	View	Download
☐		RAxML_info.result	100.94 KB	View	Download
☐		_scheduler_stderr.txt	870.00 bytes	View	Download
☐		RAxML_bipartitions.result	3.66 KB	View	Download
☐		infile.reduced	37.17 KB	View	Download
☐		term.txt	314.00 bytes	View	Download
☐		RAxML_bipartitionsBranchLabels.result	3.73 KB	View	Download
☐		infile	75.56 KB	View	Download
☐		_JOBINFO.TXT	402.00 bytes	View	Download

☐	PROCESS_OUTPUT	STDOUT	166.00 bytes	View	Download
☐		STDERR	1.28 KB	View	Download
☐	all_results	logfile.txt	38.10 KB	View	Download
☐		stderr.txt	1.28 KB	View	Download
☐		scheduler.conf	76.00 bytes	View	Download
☐		stdout.txt	166.00 bytes	View	Download
☐		start.txt	40.00 bytes	View	Download
☐		fastree_result.tree	2.40 KB	View	Download
☐		done.txt	49.00 bytes	View	Download
☐		_scheduler_stderr.txt	870.00 bytes	View	Download
☐		term.txt	314.00 bytes	View	Download
☐		infile	75.56 KB	View	Download
☐		_JOBINFO.TXT	347.00 bytes	View	Download

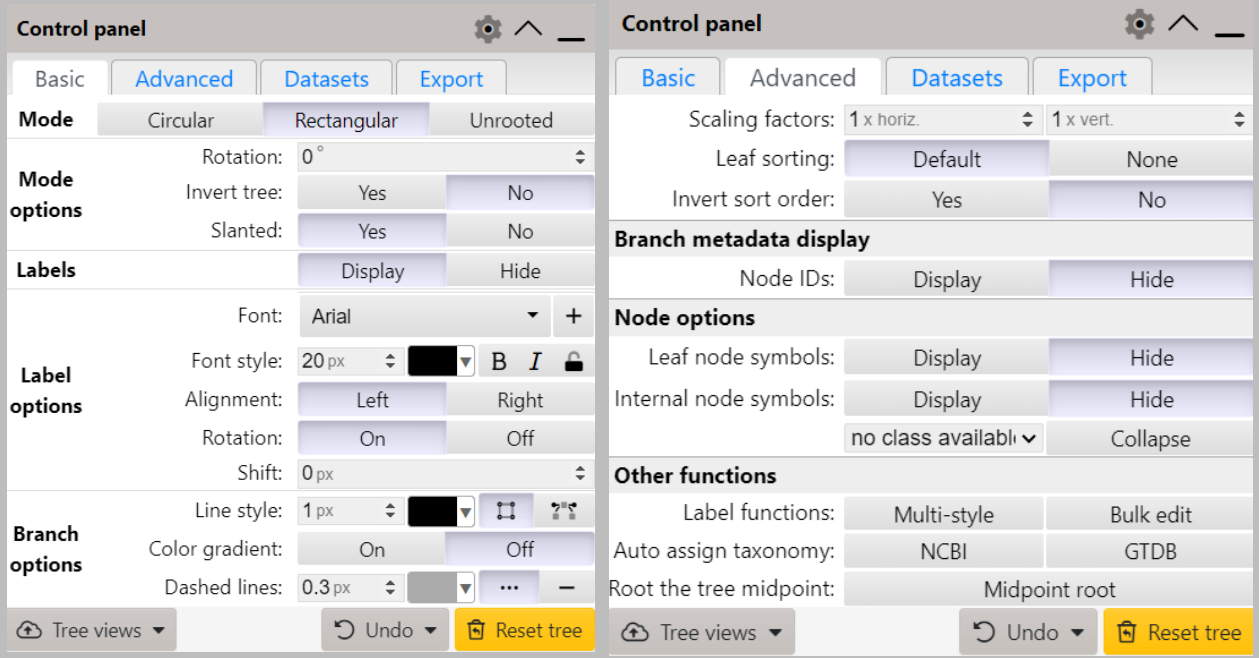
Download output data from CIPRES

Build a Tree using iTOL



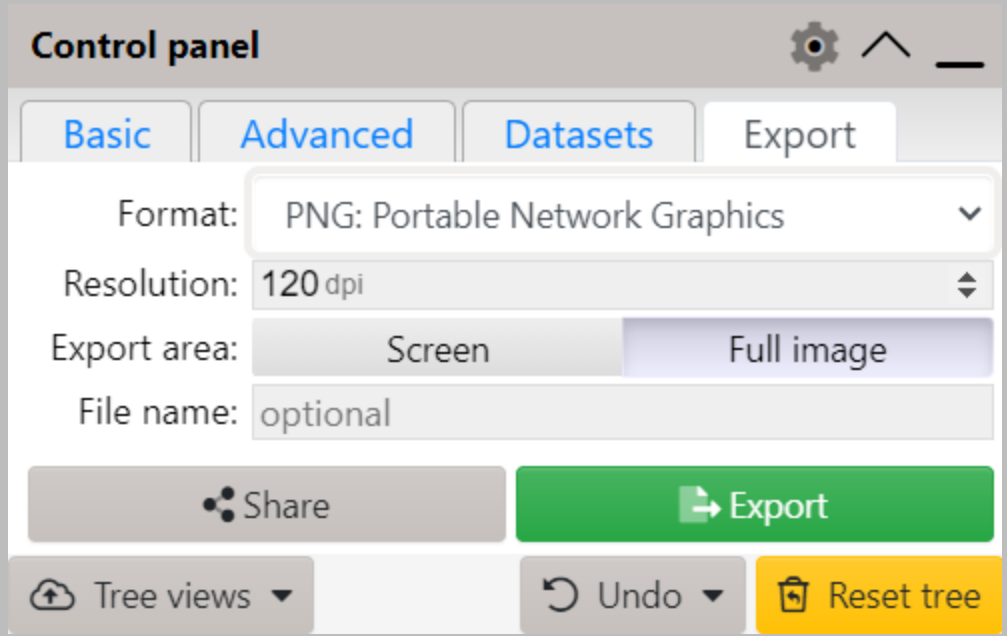
Interactive Tree of Life: https://itol.embl.de/upload.cgi

Build a Tree using iTOL



Edit the tree

Build a Tree using iTOL



Export the Tree

Q&A