# RAxML vs. FastTree

## ECES T480/680
## Winter 2017

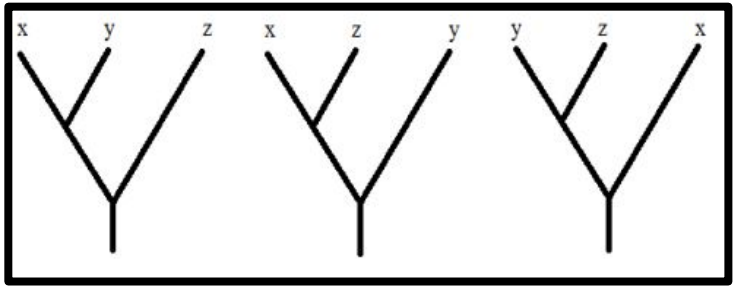
Rebecca Cargan and Bairavi Venkatesh

# Outline

I. Intro to Phylogenetics and Maximum Likelihood
II. RAxML
    A. Algorithms
    B. CIPRES Demo
    C. Proteus Demo
III. FastTree
    A. Algorithms
    B. CIPRES Demo
    C. Proteus Demo
IV. Compare Trees
    A. RaxML - 16S vs GlnS
    B. FastTree - 16S vs GlnS
V. Pros and Cons

# An Overview of Phylogenetic Analysis

- **Phylogeny** = branching diagram that reveals the evolutionary history of a group of entities
- **Phylogenetic Reconstruction** = the attempt to discern the ancestral relationships between a set of sequences
- For n species, there are $\frac{(2n-3)!}{2^{n-1}(n-1)!}$ possible trees
- With more species, computation would take years to complete
- Heuristic Methods = approximation techniques to avoid searching entire tree space.
  - Distance Methods
  - Parsimony
  - Maximum Likelihood



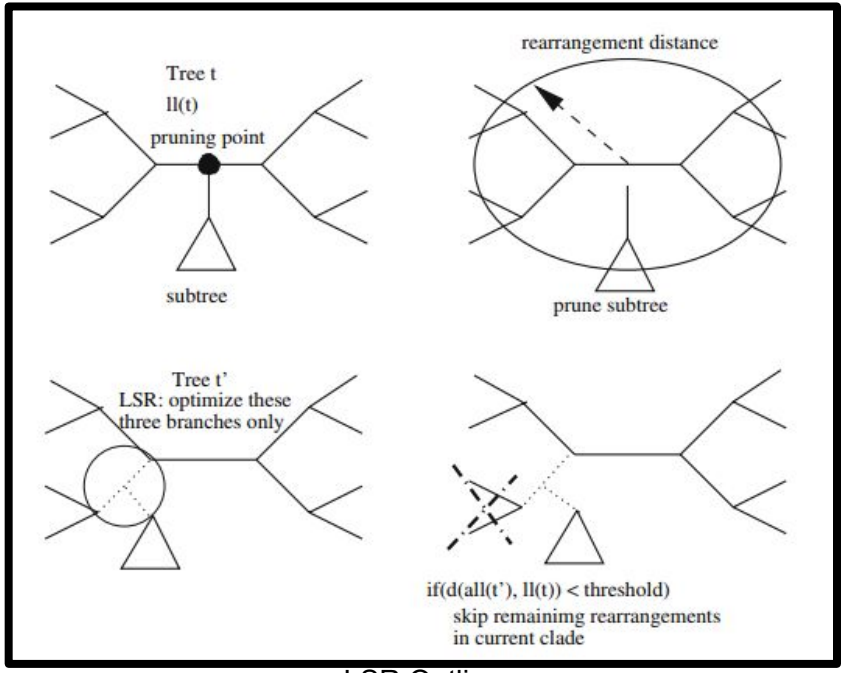

| n-species | Rooted Trees |
|---|---|
| 3 | 3 |
| 4 | 15 |
| 10 | 105 |
| 20 | 8E21 |

# Maximum Likelihood

- The process of finding the topology and branch lengths of a tree to yield the greatest probability of observing the DNA sequences in our data
- If D = data, Θ = tree, and M = model of evolution then, $\underline{\text{Likelihood} = P(D|\Theta,M)}$
- RAxML and FastTree use maximum likelihood, in addition to parsimony and distance methods to generate best tree estimate
- Basic Steps:
  - 1) Start with best first guess (using parsimony or distance method)
  - 2) Optimize Tree
  - 3) Calculate Maximum Likelihoods
  - 4) Conduct bootstrapping

# RAxML - Algorithm

- Randomized A(x)cellerated Maximum Likelihood
- Program for maximum likelihood based inference of large (1000+ taxa) phylogenetic trees
- Step 1 = Construct Parsimony Tree
  - Find the tree that requires fewest evolutionary changes
  - Randomized input sequence order results in different tree each time
  - Can be used to build consensus tree
- Step 2 = Optimize Tree
  - Lazy Subtree Rearrangement (LSR) - Pruning/regrafting subtrees to optimize topology
- Step 3 = Perform Bootstrapping
  - Add confidence levels to tree branches



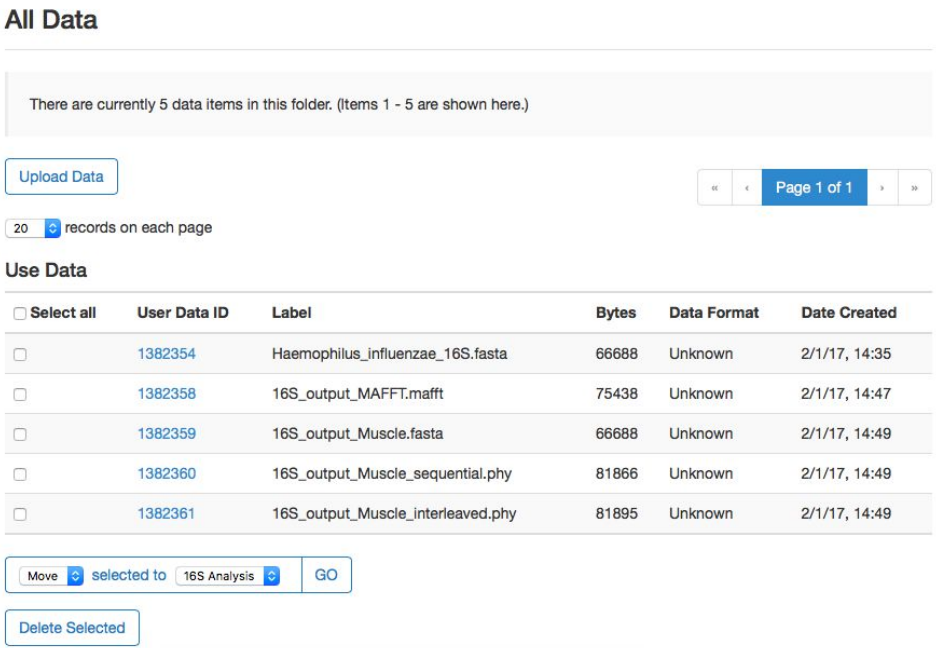

LSR Outline

DOI: 10.1007/s11265-007-0067-4

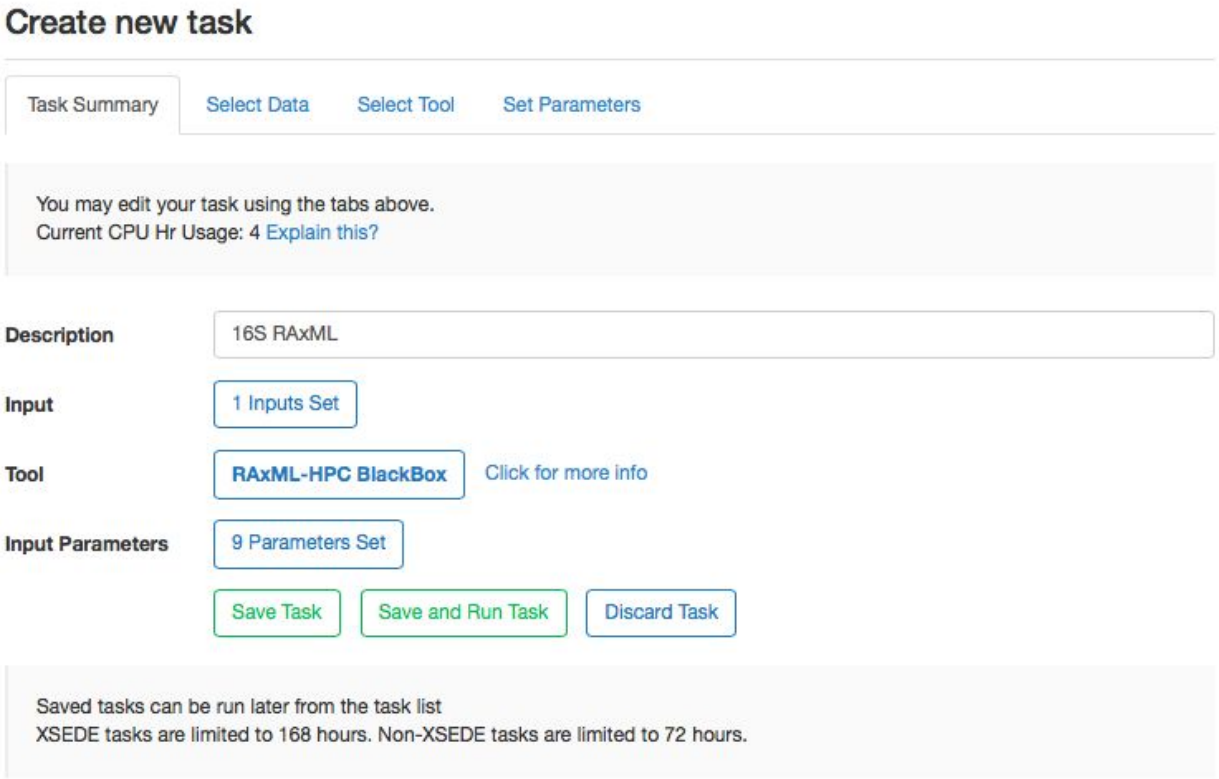
# RAxML - CIPRES Demo

Initial Dataset

Aligned Data



# RAxML - CIPRES Demo



# RAxML - CIPRES Demo

**RAxML-HPC BlackBox: Phylogenetic tree inference using maximum likelihood/rapid bootstrapping on XSEDE. (Alexandros Stamatakis)**

Simple Parameters

Maximum Hours to Run (click here for help setting this correctly) * 0.25

Sequence Type * ○ **Protein** ◉ **Nucleotide**

Outgroup (one or more comma-separated outgroups, see comment for syntax)

Constraint (-g) (file name of a multifurcating constraint tree)

Binary Backbone (-r) (file name of a binary constraint tree)

Use a mixed/partitioned model? (-q) (typically used for multi-gene alignments)

Create an input file that excludes the range of positions specifed in this file (-E)

Estimate proportion of invariable sites (GTRGAMMA + I) ◉ **no** ○ **yes**

Protein Substitution Matrix * JTT

Use empirical base frequencies? * ○ **yes** ◉ **no**

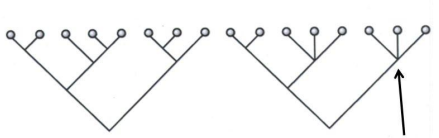
Find best tree using maximum likelihood search ☑

Let RAxML halt bootstrapping automatically (HIGHLY recommended) * ☑

Don't use BFGS searching algorithm (--no-bfgs) * ☐ (BFGS can produce ~30% speedup)

Print branch lengths (-k) ☐ (bootstrapped trees should be printed with branch lengths)

**Bifurcating vs multifurcating trees**



# RAxML - CIPRES Demo

**RAxML-HPC2 on XSEDE: Phylogenetic tree inference using maximum likelihood/rapid bootstrapping run on XSEDE (Alexandros Stamatakis)**

Simple Parameters

Maximum Hours to Run (click here for help setting this correctly) * 0.25

Set a name for output files * result

Enable ML searches under CAT (-F) *

I have a data set that may require more than 15 GB of memory *

Enter the number of patterns in your dataset

Enter the number of taxa in your dataset

Please select the Data Type * Nucleotide

- Protein
- ✓ Nucleotide
- RNA Structure
- Binary Morphological
- Multi-State Morphological

Outgroup (one or more comma-separated outgroups, see comment for syntax)

Specify the number of distinct rate categories (-c) * 25

Disable Rate Heterogeneity (-V) *

Supply a tree (Not available when doing rapid bootstrapping, -x) (-t)

Specify a random seed value for parsimony inferences (-p) *

Enter a random seed value for parsimony inferences (gives reproducible results from random starting tree) * 12345

Specify an initial rearrangement setting (-i) *

Specify the distance from original pruning point (-i) * 10

Constraint (-g)

Binary Backbone (-r)

Use a mixed/partitioned model? (-q)

Estimate individual per-partition branch lengths (-M) *

Correct for Ascertainment bias (ASC_) no yes

Ascertainment bias correction type (--asc-corr) * Lewis Felsenstein Stamatakis

Estimate proportion of invariable sites (GTRGAMMA + I) * yes no

Choose an input file that excludes the range of positions specifed in this file (-E)

Weight characters as specifed in this file (-a)

Disable checking for sequences with no values (-O)

Print output files that can be parsed by Mesquite. (-mesquite)

## Select the Analysis

Only compute a randomized parsimony starting tree (-y)

Specify the number alternative runs on distinct starting trees? (-#/-N) +

Enter number of number alternative runs +

Don't use BFGS searching algorithm (--no-bfgs)

Draw bipartitions onto a single tree topology. (-f b)

Compute Marginal Ancestral States using a rooted reference tree. (-f A)

Compute a log likelihood test (-f h)

Do A Final Opimization of ML Tree (-f T)

Write intermediate tree files to a file (-j)

Use ML search convergence criterion. (-D)

Compute majority rule consensus tree (-J) +

Specify majority rule consensus tree (-J) technique + Majority rule

File with topologies for bipartitions or bootstopping (-z)

Compute pair-wise ML distances (-f x; GAMMA models only) +

Run very fast experimental tree search(-f E) +

Execute morphological weight calibration using maximum likelihood (-f u) +

Classify a bunch of environmental sequences into a reference tree using thorough read insertions(-f v) +

## Configure Bootstrapping

Conduct Multiparametric Bootstrapping? (-b) +

Enter a random seed value for multi-parametric bootstrapping + 12345

Conduct rapid bootstrapping? (-x)

Enter a random seed value for rapid bootstrapping + 12345

Conduct a rapid Bootstrap analysis and search for the best-scoring ML tree in one single program run. (-f a)

Print branch lengths (-k)

Specify an Explicit Number of Bootstraps +

Bootstrap iterations (-#|-N) + 100

Let RAxML halt bootstrapping automatically +

Stop Bootstrapping Automatically with Frequency Criterion +

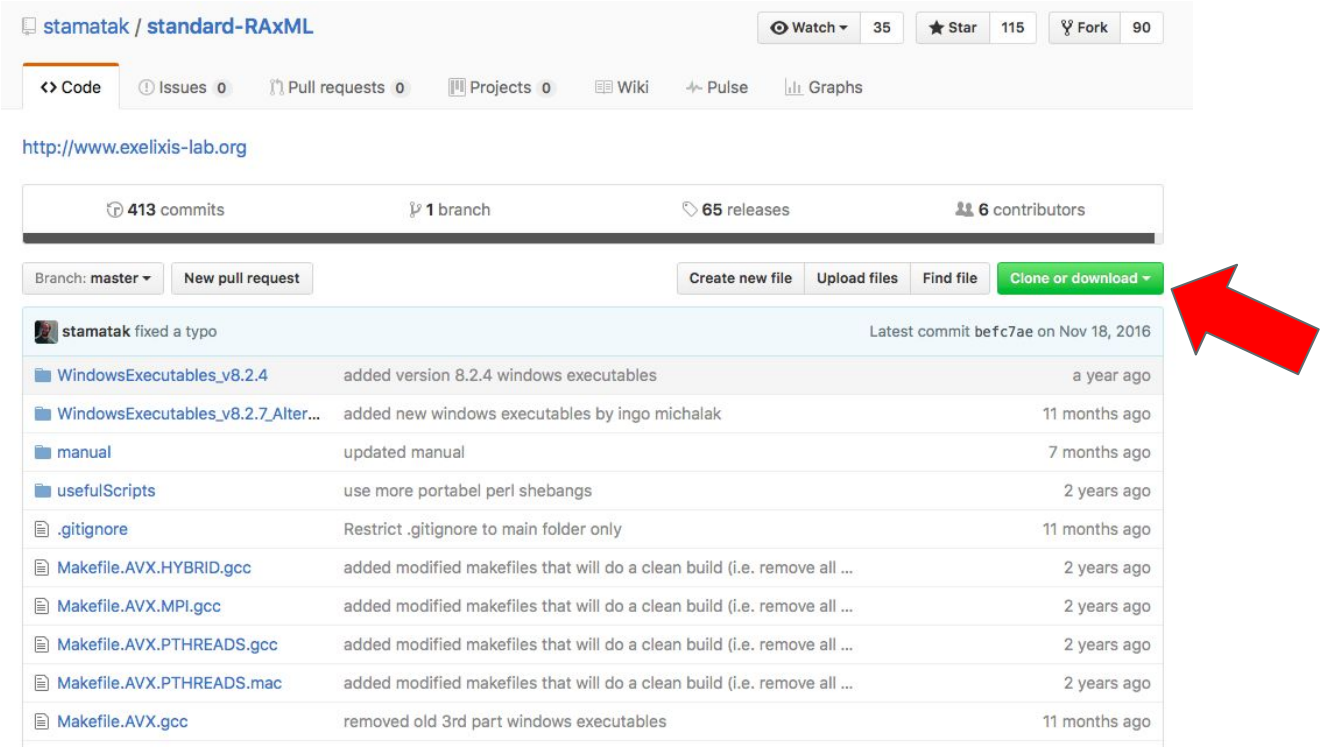
Stop Bootstrapping Automatically with Majority Rule Criterion (recommended) +

Select Majority Rule Criterion: (autoMRE is recommended) + autoMR autoMRE autoMRE_IGN

Use a posteriori bootstrapping

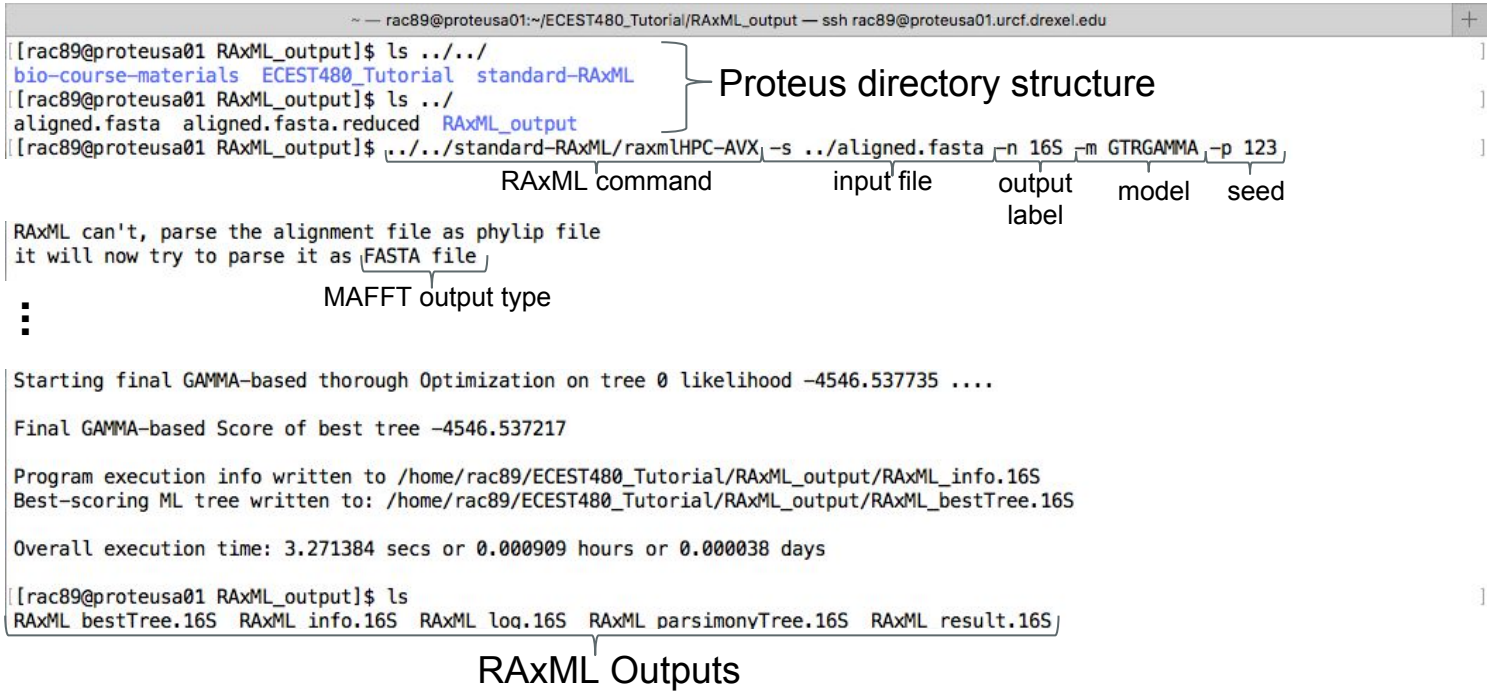
Select the criterion for a posteriori bootstrapping analysis + autoFC

# RAxML - Proteus Demo



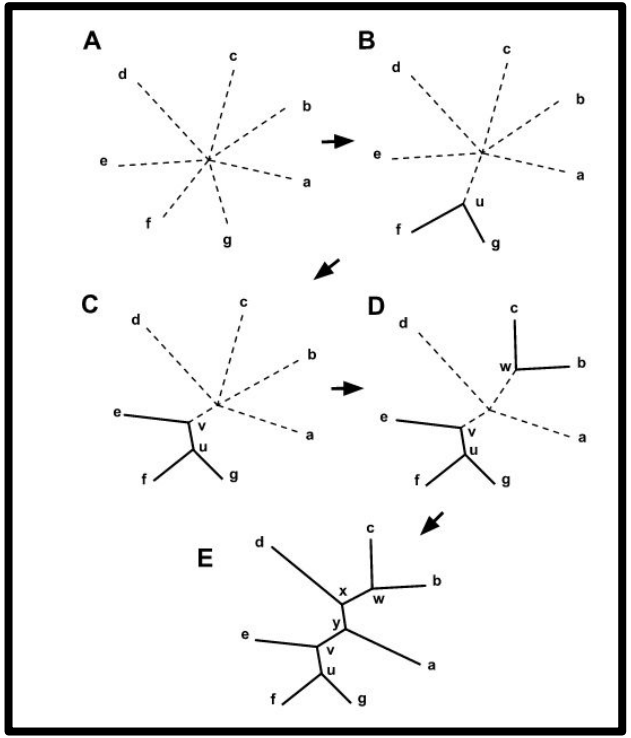
# RAxML - Proteus Demo

- To install, run “make -f Makefile.xxx.gcc” within RAxML Standard Directory
- Run RAxML command from the directory where you want the outputs



# FastTree - Algorithm

- A tool for inferring Maximum Likelihood trees for many (1000+ taxa) alignments.
- Step 1= Heuristic Neighbor Joining
  - Keeps track of internal nodes rather than distance matrix
- Step 2= Minimum Evolution
  - Reduce the sum of branch lengths
  - Nearest Neighbor Interchange
- Step 3= Bootstrapping
  - Shimodaira-Hasegawa method
  - “Fast and global”

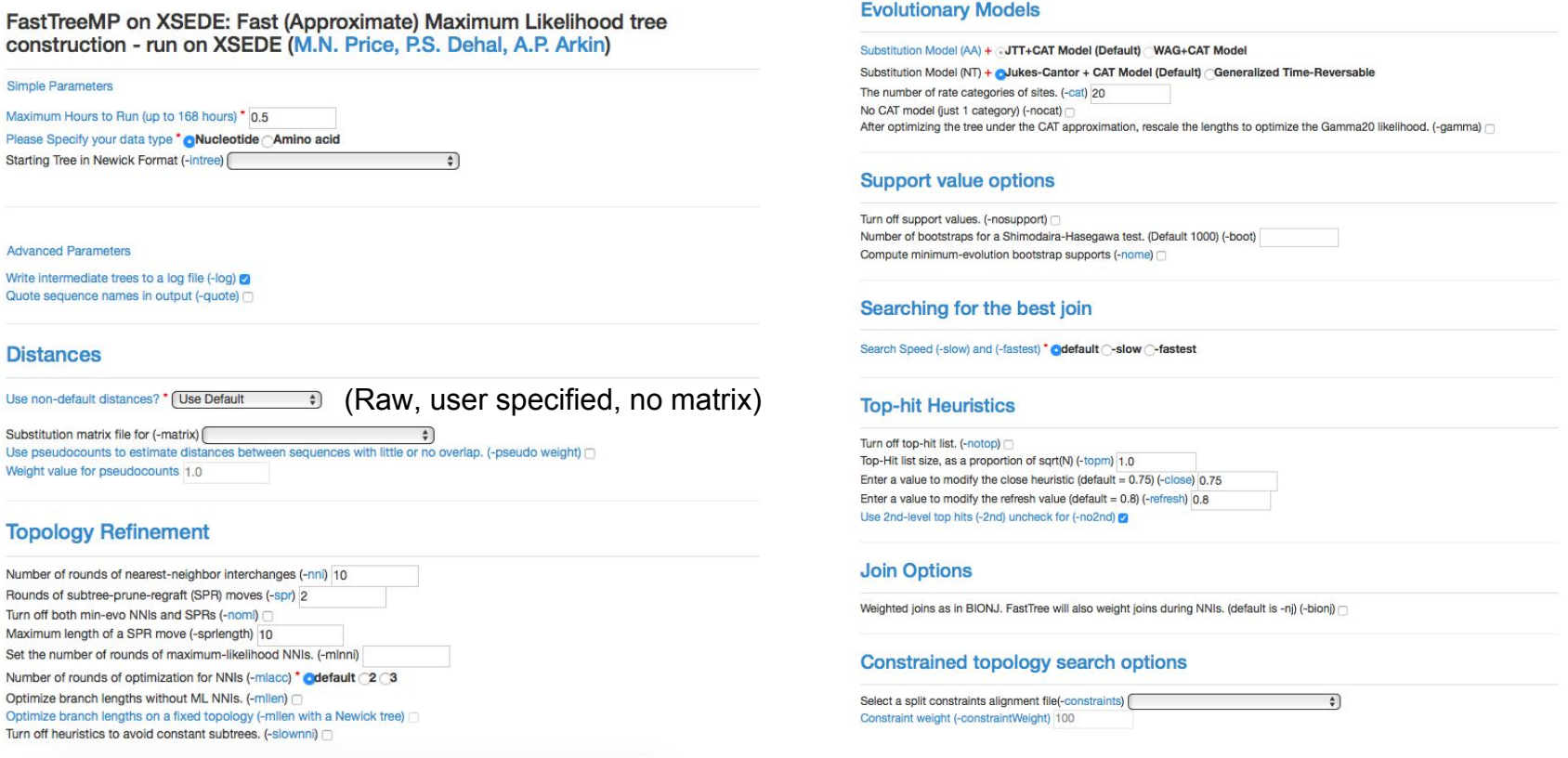


Standard Neighbor Joining Method

Neighbor Joining → $d(FG,E) = \frac{d(F,E) + d(G,E)}{2}$

FastTree → $d(FG,E) = \Delta(FG,E) - u(E) - u(FG)$

# FastTree – CIPRES Demo



**FastTreeMP on XSEDE: Fast (Approximate) Maximum Likelihood tree construction - run on XSEDE (M.N. Price, P.S. Dehal, A.P. Arkin)**

Simple Parameters

Maximum Hours to Run (up to 168 hours) * 0.5

Please Specify your data type * Nucleotide Amino acid

Starting Tree in Newick Format (-intree)

Advanced Parameters

Write intermediate trees to a log file (-log)

Quote sequence names in output (-quote)

**Distances**

Use non-default distances? * Use Default (Raw, user specified, no matrix)

Substitution matrix file for (-matrix)

Use pseudocounts to estimate distances between sequences with little or no overlap. (-pseudo weight)

Weight value for pseudocounts 1.0

**Topology Refinement**

Number of rounds of nearest-neighbor interchanges (-nni) 10

Rounds of subtree-prune-regraft (SPR) moves (-spr) 2

Turn off both min-evo NNIs and SPRs (-nomi)

Maximum length of a SPR move (-sprlength) 10

Set the number of rounds of maximum-likelihood NNIs. (-mlnni)

Number of rounds of optimization for NNIs (-mlacc) * default 2 3

Optimize branch lengths without ML NNIs. (-mllen)

Optimize branch lengths on a fixed topology (-mllen with a Newick tree)

Turn off heuristics to avoid constant subtrees. (-slownni)

**Evolutionary Models**

Substitution Model (AA) + JTT+CAT Model (Default) WAG+CAT Model

Substitution Model (NT) + Jukes-Cantor + CAT Model (Default) Generalized Time-Reversable

The number of rate categories of sites. (-cat) 20

No CAT model (just 1 category) (-nocat)

After optimizing the tree under the CAT approximation, rescale the lengths to optimize the Gamma20 likelihood. (-gamma)

**Support value options**

Turn off support values. (-nosupport)

Number of bootstraps for a Shimodaira-Hasegawa test. (Default 1000) (-boot)

Compute minimum-evolution bootstrap supports (-nome)

**Searching for the best join**

Search Speed (-slow) and (-fastest) * default -slow -fastest

**Top-hit Heuristics**

Turn off top-hit list. (-notop)

Top-Hit list size, as a proportion of sqrt(N) (-topm) 1.0

Enter a value to modify the close heuristic (default = 0.75) (-close) 0.75

Enter a value to modify the refresh value (default = 0.8) (-refresh) 0.8

Use 2nd-level top hits (-2nd) uncheck for (-no2nd)

**Join Options**

Weighted joins as in BIONJ. FastTree will also weight joins during NNIs. (default is -nj) (-bionj)

**Constrained topology search options**

Select a split constraints alignment file(-constraints)

Constraint weight (-constraintWeight) 100

# FastTree - Proteus Demo

**Running FastTree**

To infer a tree for a protein alignment with the JTT+CAT model, use

```
FastTree < alignment_file > tree_file
```

or

```
FastTree alignment.file > tree_file
```

Use the **-wag** or **-lg** options to use the WAG+CAT or LG+CAT model instead.

To infer a tree for a nucleotide alignment with the GTR+CAT model, use

```
FastTree -gtr -nt < alignment.file > tree_file
```

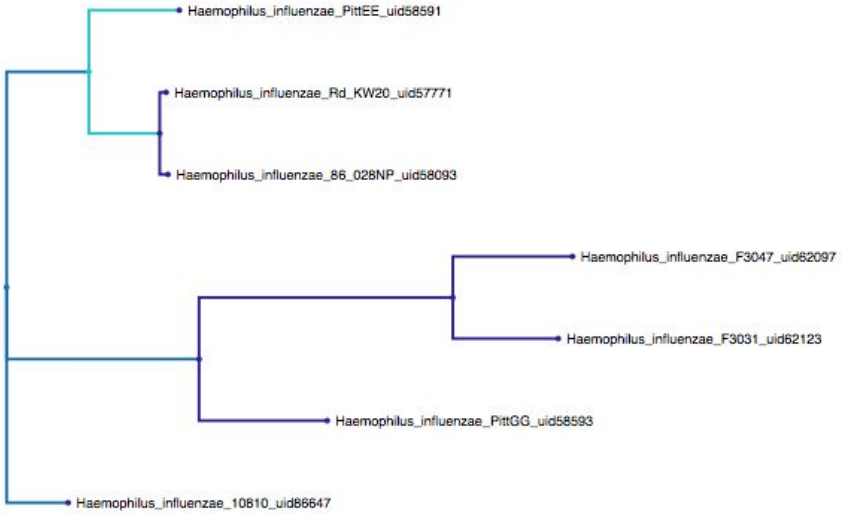
or

```
FastTree -gtr -nt alignment_file > tree_file
```

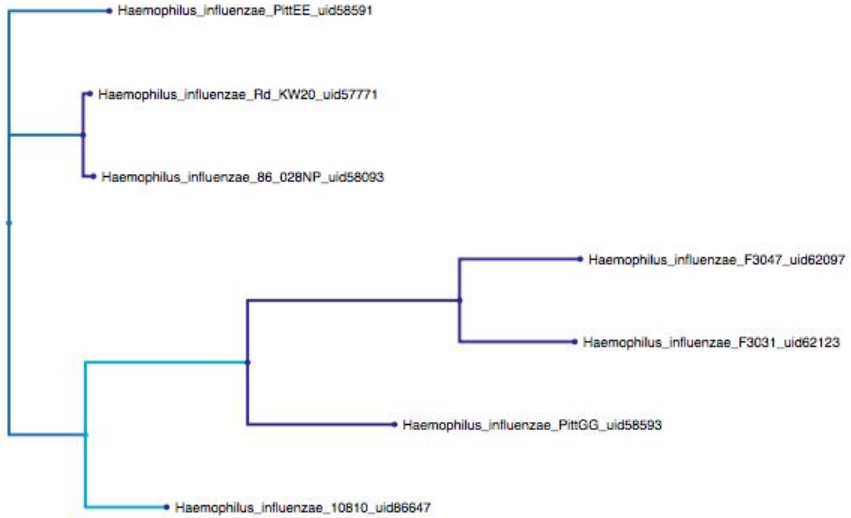
If you do not specify **-gtr**, then FastTree will use the Jukes-Cantor + CAT model instead.

Use the **-gamma** option (about 5% slower) if you want to rescale the branch lengths and compute a Gamma20-based likelihood. Gamma likelihoods are more comparable across runs. These also allow for statistical comparisons of the likelihood of different topologies if you use the **-log logfile** option (see details). The change in the scale of the tree is usually modest (10% or less).

# GlnS: RAxML vs FastTree Comparison

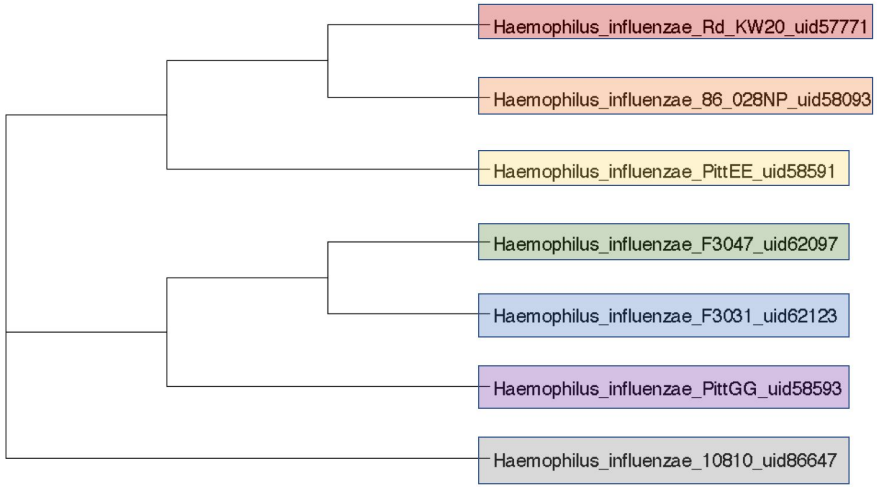



RAxML

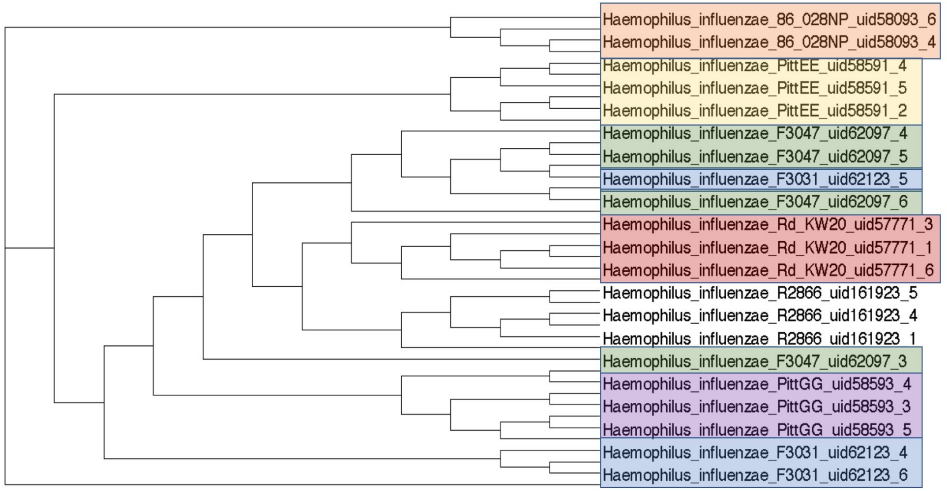



FastTree

# RAxML: 16S vs GlnS Phylogenetic Trees

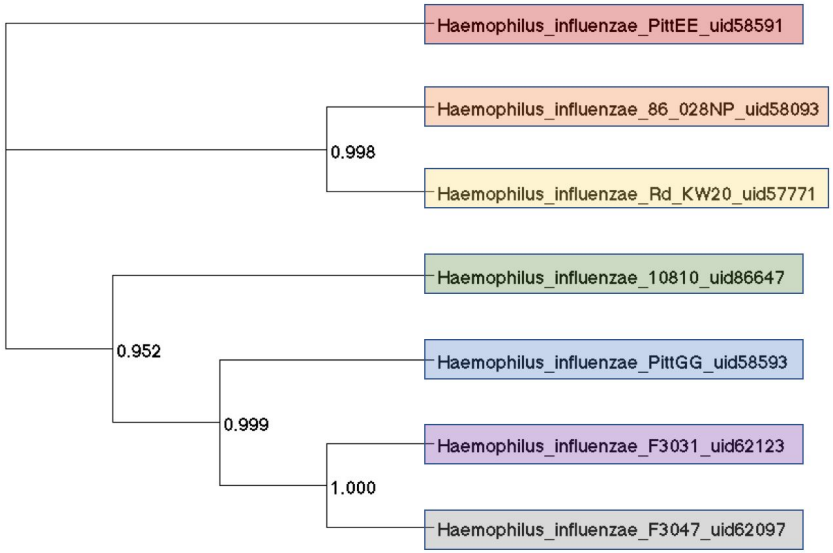



GlnS

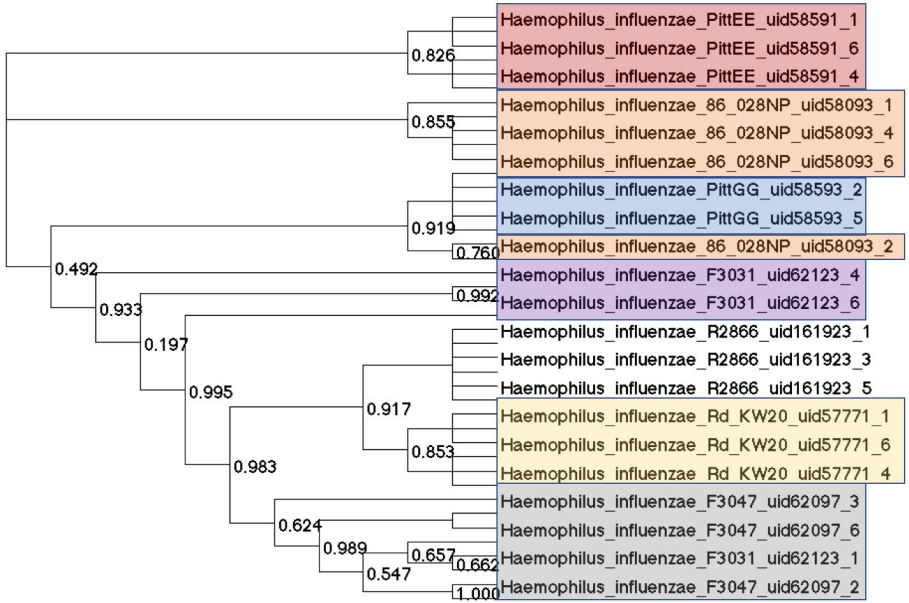



16S

# FastTree: 16S vs GlnS Phylogenetic Trees




GlnS




16S

# RAxML vs. FastTree

## RAxML

- Due to Subtree Pruning Method, all individual subtrees and the changes in topology must be kept in memory
- Produces more accurate ML values
- Lower accuracy on large datasets
- Very thorough manual with descriptions of options
- “Black Box” version on Cipres
- Interdependent options require careful command line usage
- Can run upon compile (using makefile), no formal installation needed

## FastTree

- Only considers Nearest Neighbor Interchanges, therefore memory of changing topologies don’t need to be considered
- Produces more accurate tree topology
- Better accuracy on large datasets, even with errors in alignment
- “Plug-and-play” executable
  - Full installation necessary on Mac
- Options are abstracted in command line
- Parameters are easy to set on Cipres

# Resources

[1] A. Stamatakis et al. Exploring new search algorithms and hardware for phylogenetics: RAxML meets the IBM cell. The Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology 48(3), pp. 271-286. 2007. . DOI: 10.1007/s11265-007-0067-4

[2] M. N. Price, P. S. Dehal and A. P. Arkin. FastTree: Computing large minimum evolution trees with profiles instead of a distance matrix. Molecular Biology and Evolution 26(7), pp. 1641-1650. 2009. . DOI: 10.1093/molbev/msp077

[3] K. Liu, C. R. Linder and T. Warnow. RAxML and FastTree: Comparing two methods for large-scale maximum likelihood phylogeny estimation. PloS One 6(11), pp. e27731. 2011. . DOI: 10.1371/journal.pone.0027731.

[4] Lawrence Berkeley National Lab, FastTree Manual., http://www.microbesonline.org/fasttree/

[5] Stamatakis A., The RAxML v8.2.X Manual., http://sco.h-its.org/exelixis/resource/download/NewManual.pdf