Phylogenetics

Introduction to Bioinformatics
Dortmund, 16.-20.07.2007

Lectures:
Sven Rahmann

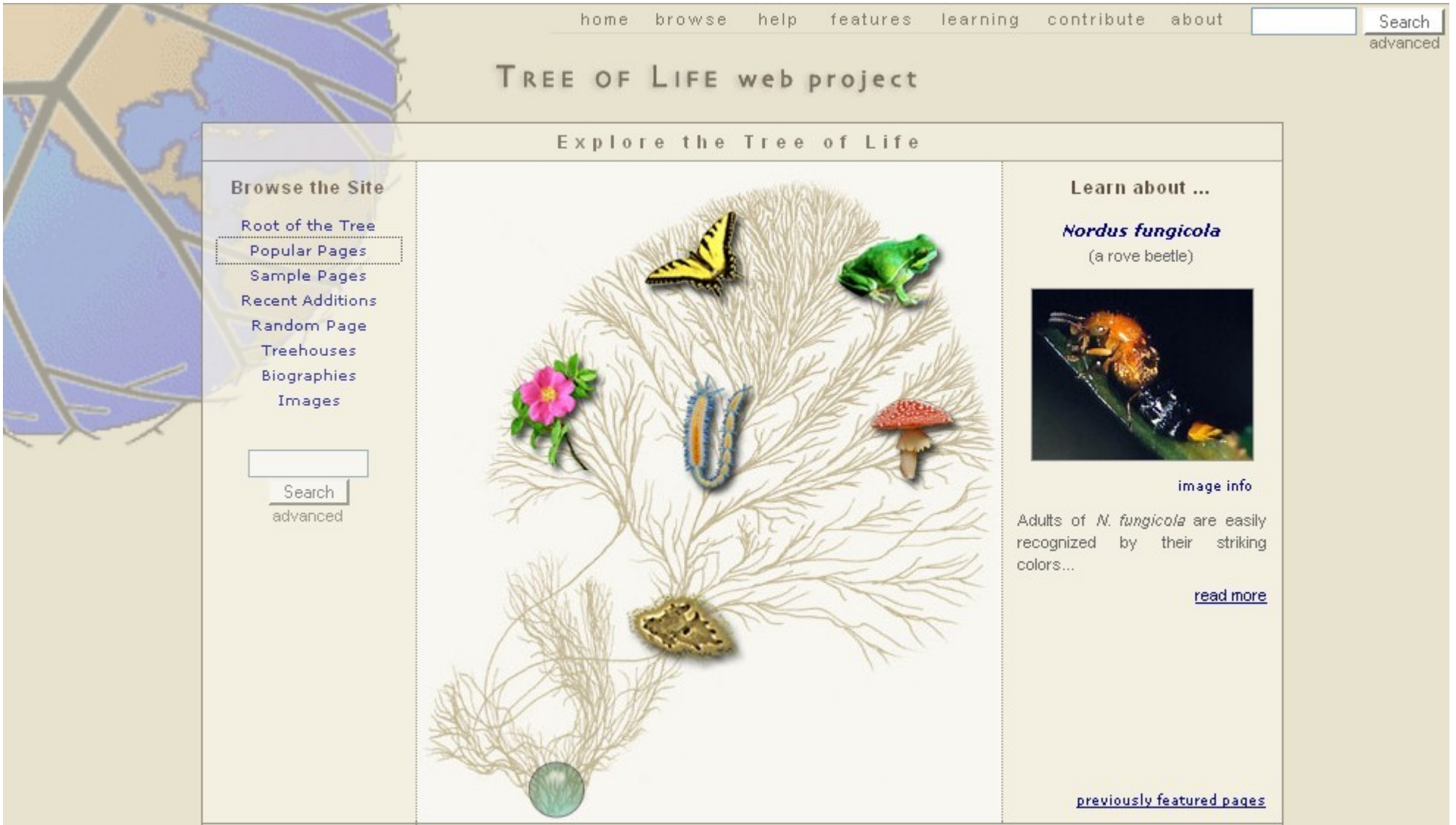
Exercises:
Udo Feldkamp, Michael Wurst

Phylogenetics

- phylum = tree
- phylogenetics:
  reconstruction of evolutionary trees
- phylogeny:
  an evolutionary tree, “Stammbaum”

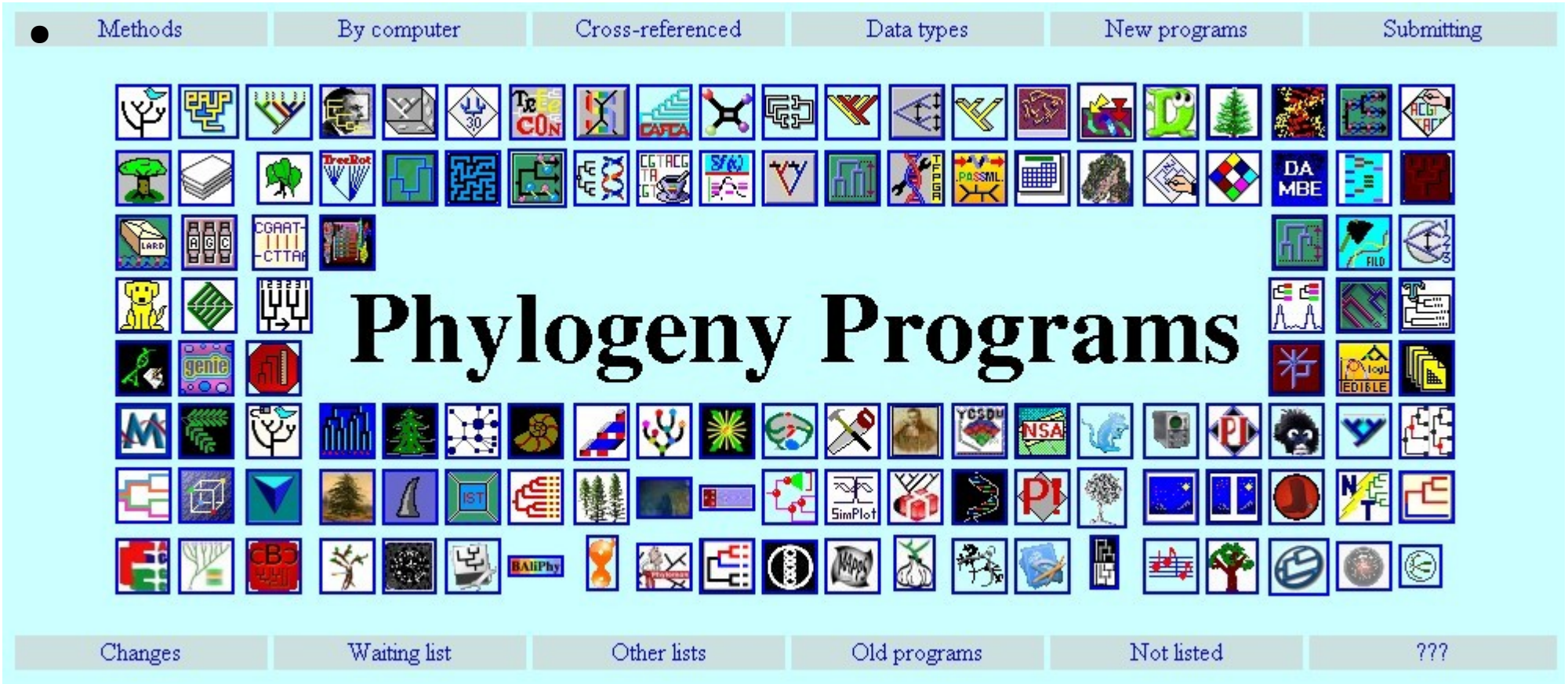
Tree Of Life Web Project

URL: http://www.tolweb.org



Software Collection

- URL: http://evolution.genetics.washington.edu/phylip/software.html



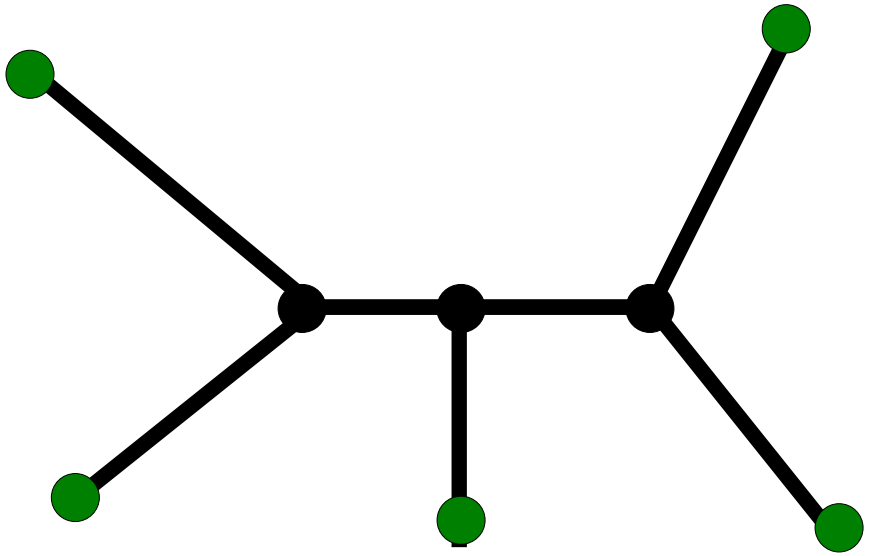
PHYLIP

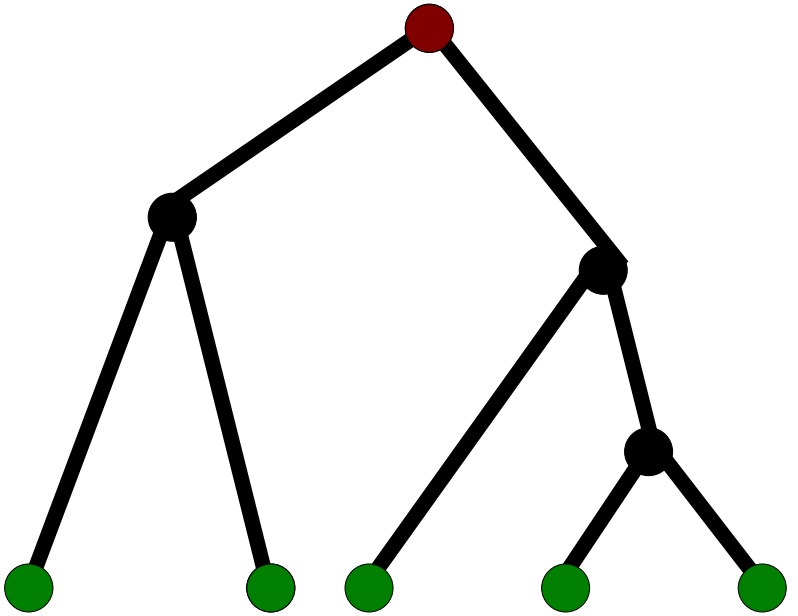
- PHYLIP is one of the most widely used software packages for phylogenetic analysis.
- PHYLIP project homepage: http://evolution.genetics.washington.edu/phylip.html
- Online server URL: http://bioweb.pasteur.fr/seqanal/phylogeny/phylip-uk.html

Trees

- T = (V,E)
  a **tree** is a **graph**, consists of **vertices** and **edges**
- V = vertices, also called nodes
  - leaves L, inner nodes N, root r (for rooted trees)
- E = edges (connect vertices)
- Trees can be **rooted** or **unrooted**
- Trees are **connected**, **acyclic** graphs
- Unrooted binary trees satisfy:
  $|E| = 2|L| - 3$ and $|N| = |L| - 2$
  Rooted trees have one more edge, plus a root

Unrooted and Rooted Trees





inner node

leaf

root

edge

Number of Trees

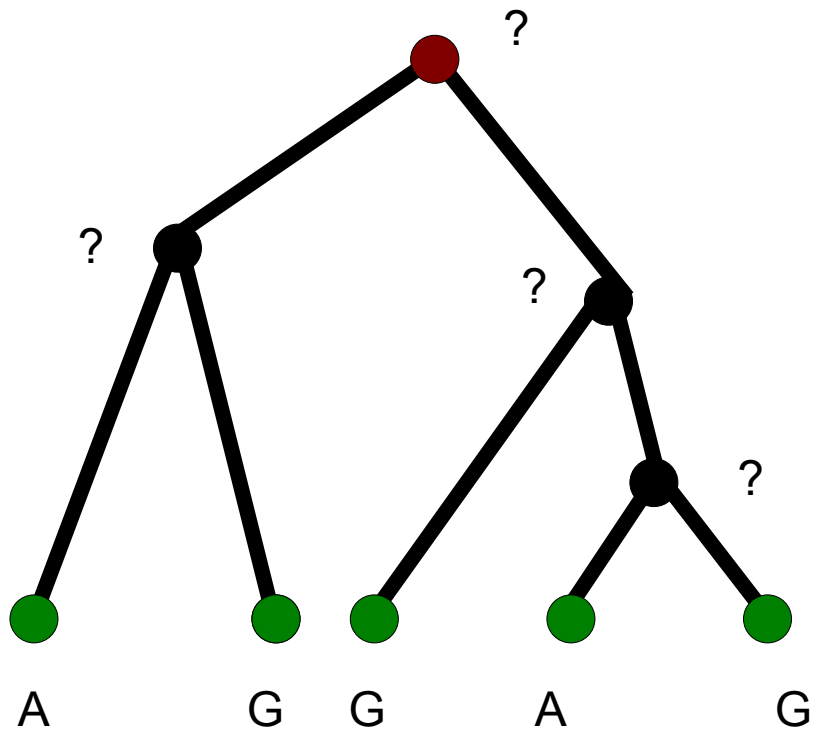
- unrooted trees
  3 leaves: 1
  4 leaves: 3
  5 leaves: 3*5 = 15
  6 leaves: 15*7 = 105
- rooted trees
  3 leaves: 3
  4 leaves: 3*5 = 15
  5 leaves: 15*7 = 105
- super-exponentially many trees

Principles in Phylogenetics

- **Parsimony Methods**:
  Occam's razor,
  simplest (=shortest) explanation is best
- **Distance-based methods**:
  Distances in tree should resemble pairwise distances between sequences
- **Maximum Likelihood methods**:
  what's the most plausible (not: probable!) evolutionary scenario?
- **Bayesian methods**:
  what's the most probable scenario, considering prior knowledge and the sequence data?

Small Parsimony Problem

- Given a tree, sequences at the leaves, a multiple alignment, a cost matrix for substitutions,
- find sequences at inner nodes of the tree to minimize overall change cost along all edges
- Efficient algorithms:
  - Fitch O(|V|*|Alphabet|)
  - Sankoff



Big Parsimony Problem

- Given sequences (at the leaves),
- find a tree and the best labeling of inner nodes with minimal substitution cost
- No efficient algorithm known,
  problem is NP-hard
- Essentially have to enumerate all trees.
  Super-exponentially many trees!

Distance-Based Methods

- Given sequences,
  first compute a pairwise distance matrix using
  - edit distance
  - edit distance “corrected” for minimality
    (“Jukes-Cantor correction”, “Kimura correction”)
  - distance based on an evolutionary model based on a time-continuous Markov process
- Then find a **tree** (unrooted or rooted) and **edge lengths**, such that distances in the tree match all pairwise distances in the distance matrix

Fitting Distances on a Tree: Problems

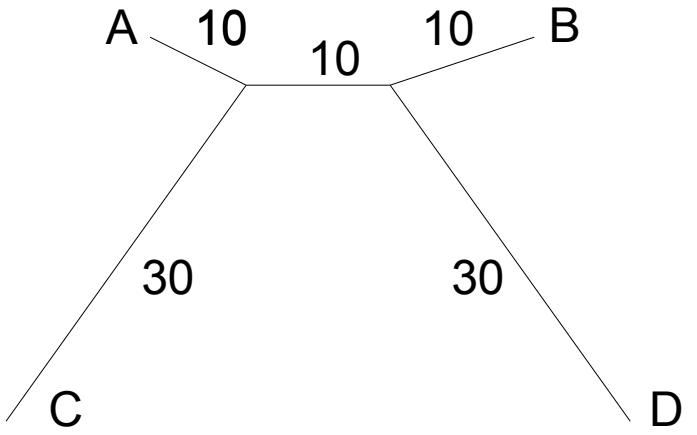
- More pairwise distance values in the matrix than edges in the tree: problem overdetermined.
  A perfectly fitting tree does not always exist!
- **Metric**: typical distance properties
  Tree metric: Distance values fit a tree
  Ultrametric: Distance values fit a rooted tree, all paths from root to leaves have same length
- Good algorithms:
  Find correct tree + edge lengths if one exists, find good approximation otherwise
- UPGMA for ultrametric, NJ for tree metric

Clustering Algorithm “UPGMA”

- Unweighted pair group method using averages
- always returns a rooted ultrametric tree (all leaves have same distance from the root)
- correct tree returned if distances are ultrametric
- Algorithm:
  - While more than one object remains:
    - Find the pair of objects x,y with the smallest distance
    - Replace them with a single object (x,y)
    - Compute distances from (x,y) to other objects a,b,c... by averaging d(x,a) with d(y,a), ...
  - Order in which objects are grouped defines a tree

Neighbor Joining (NJ)

- creates an unrooted tree by iteratively joining two subtrees, taking into account their distance and also the distance between all other subtrees.
- The two closest sequences need not be neighbors!



d(A,B) = 30, smallest;
but tree is AC || BD

- NJ finds the correct tree if the distances admit one.
- NJ finds a “good” tree otherwise (heuristic)

Probabilistic Methods

- Require a model of an evolutionary process, sometimes limited to substitutions (no gaps)
- Maximum Likelihood (ML)
  - Assuming a tree topology and edge lengths (T,L), what is the total probability P(seqs | T,L), summed over all choices of inner node sequences, that this choice generates the observed sequences?
    This is the Likelihood of (T,L)
    Maximize this over all possible choices (T,L)
- Bayesian (more natural question)
  - Using prior knowledge / personal bias, for each choice (T,L) as above, compute P((T,L) | seqs), conditional probability of (T,L), given the seqs.

Probabilistic Models

- require understanding of time-continuous Markov processes as evolutionary substitution models
- require understanding of probability theory
- Top-level view: a similar, but “softer” approach, than Parsimony methods. There is not “one” solution, but each tree has a certain likelihood / probability.
- No details on algorithms given here

Which Method should I use? (Personal Opinion)

- Distance-Based methods usually work fine.
  As a good first choice, run some NJ variant.
- Parsimony may underestimate the true number of evolutionary changes, as it looks for the "shortest" possible explanation.
  OK when sequences are closely related.
  Problem when sequences are distantly related!
  Parsimony has no "edge lengths"!
- Probabilistic methods might return more accurate trees than distance methods,
  but are usually slower.

How robust is the tree?

- Robustness :=
  tree does not change (fundamentally) when small errors are introduced into the data
- Robustness is not accuracy!
- Accuracy :=
  the tree is (close to) the biologically correct one
- A tree that is not robust, however, is “instable”, and unlikely to be accurate.
- Accuracy: hard to measure (except in simulations)
- Robustness: easy to measure

Measuring Robustness

- Basic idea:
  - For as many times as possible,
    - modify original sequences / alignment slightly
    - compute and store tree for modified data
  - Finally, compare original tree with those trees
  - Or, compute a consensus tree (multifurcating?)
- Frequently done using “bootstrap”:
  - Randomly draw a selection of original alignment columns, of the same cardinality as original alignment
- Phylip contains a program for generating bootstrap trees.