MCB 372

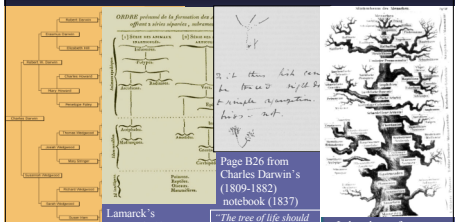
Trees

Phylogenetic reconstruction

PHYLIP

Peter Gogarten
Office: *BSP 404*
phone: *860 486-4061,*
Email: gogarten@uconn.edu

Trees as a Tool to Visualize Evolutionary History



Page B26 from Charles Darwin's (1809-1882) notebook (1837)

Lamarck's "Tree of Life" (1815)

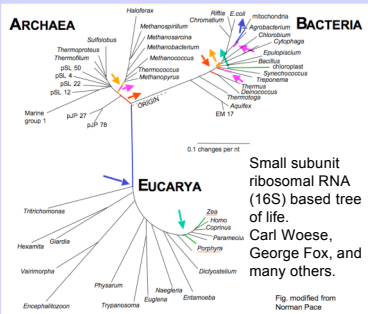
"The tree of life should perhaps be called the coral of life, base of branches dead"

Lebensbaum from Ernst Haeckel, 1874

Family trees (Charles Darwin http://www.aboutdarwin.com/)

PHYLOGENY: from Greek phylon, race or class, and -geneia, born.
"*the origin and evolution of a set of organisms, usually of a species*" (Wikipedia);

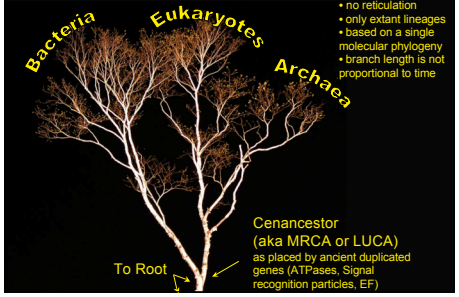
SSU-rRNA Tree of Life



Small subunit ribosomal RNA (16S) based tree of life.
Carl Woese, George Fox, and many others.

Fig. modified from Norman Pace

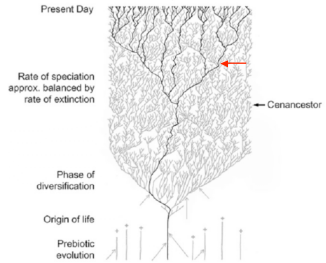
The Tree of Life according to SSU ribosomal RNA (+)



- strictly bifurcating
- no reticulation
- only extant lineages
- based on a single molecular phylogeny
- branch length is not proportional to time

Cenancestor
(aka MRCA or LUCA)
as placed by ancient duplicated genes (ATPases, Signal recognition particles, EF)

The Coral of Life (Darwin)



– the process of tracing lineages backwards in time to their common ancestors. Every two extant lineages coalesce to their most recent common ancestor. Eventually, all lineages coalesce to the cenancestor.

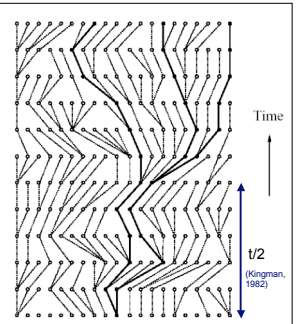
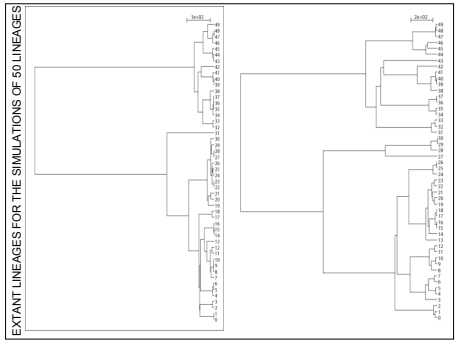
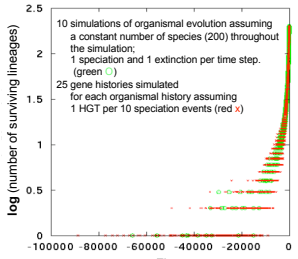


Illustration is from J. Felsenstein, "Inferring Phylogenies", Sinauer, 2003

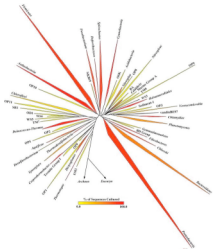


Lineages Through Time Plot



10 simulations of organismal evolution assuming a constant number of species (200) throughout the simulation;
1 speciation and 1 extinction per time step. (green ○)
25 gene histories simulated for each organismal history assuming 1 HGT per 10 speciation events (red x)

green: organismal lineages ;
red: molecular lineages (with gene transfer)



Bacterial 16SrRNA based phylogeny
(from P. D. Schloss and J. Handelsman, Microbiology and Molecular Biology Reviews, December 2004.)

The deviation from the "long branches at the base" pattern could be due to
- under sampling
- an actual radiation
  - due to an invention that was not transferred
  - following a mass extinction

What is in a tree?

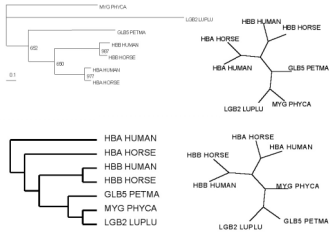
Trees form molecular data are usually calculated as unrooted trees (at least they should be - if they are not this is usually a mistake).
To root a tree you either can assume a molecular clock (substitutions occur at a constant rate, again this assumption is usually not warranted and needs to be tested),
or you can use an outgroup (i.e. something that you know forms the deepest branch).

For example, to root a phylogeny of birds, you could use the homologous characters from a reptile as outgroup; to find the root in a tree depicting the relations between different human mitochondria, you could use the mitochondria from chimpanzees or from Neanderthals as an outgroup; to root a phylogeny of alpha hemoglobins you could use a beta hemoglobin sequence, or a myoglobin sequence as outgroup.

Trees have a branching pattern (also called the **topology**), and **branch lengths**.

Often the branch lengths are ignored in depicting trees (these trees often are referred to as cladograms - note that cladograms should be considered rooted). You can swap branches attached to a node, and in an unrooted you can depict the tree as rooted in any branch you like without changing the tree.

Test:Which of these trees is different?



More tests here

Terminology

- Branches, splits, bipartitions
- In a rooted tree: clades
- Mono-, Para-, polyphyletic groups, cladists and a natural taxonomy

The term cladogram refers to a strictly bifurcating diagram, where each clade is defined by a common ancestor that only gives rise to members of this clade. I.e., a clade is monophyletic (derived from one ancestor) as opposed to polyphyletic (derived from many ancestors). (note you need to know where the root is!)

A clade is recognized and defined by shared derived characters (= synapomorphies). Shared primitive characters (= symplesiomorphies , alternativie spelling is symplesiomorphies) do not define a clade. (see in class example drawing ala Hennig).

To use these terms you need to have polarized characters; for most molecular characters you don't know which state is primitive and which is derived (exceptions:....).

Terminology

Related terms:

autapomorphy = a derived character that is only present in one group; an autapomorphic character does not tell us anything about the relationship of the group that has this character ot other groups.

homoplasy = a derived character that was derived twice independently (convergent evolution). *Note that the characters in question might still be homologous (e.g. a position in a sequence alignment, frontlimbs turned into wings in birds and bats).*

paraphyletic = a taxonomic group that is defined by a common ancestor, however, the common ancestor of this group also has decendants that do not belong to this taxonomic group. Many systematists despise paraphyletic groups (and consider them to be polyphyletic). Examples for paraphyletic groups are reptiles and protists. Many consider the archaea to be paraphyletic as well.

holophyletic = same as above, but the common ancestor gave rise only to members of the group.

homology

Two sequences are homologous, if there existed an ancestral molecule in the past that is ancestral to both of the sequences

Types of Homology

Orthologs: "deepest" bifurcation in molecular tree reflects speciation.
These are the molecules people interested in the taxonomic classification of organisms want to study.

Paralogs: "deepest" bifurcation in molecular tree reflects gene duplication. The study of paralogs and their distribution in genomes provides clues on the way genomes evolved. Gen and genome duplication have emerged as the most important pathway to molecular innovation, including the evolution of developmental pathways.

Xenologs: gene was obtained by organism through horizontal transfer. The classic example for Xenologs are antibiotic resistance genes, but the history of many other molecules also fits into this category: inteins, selfsplicing introns, transposable elements, ion pumps, other transporters,

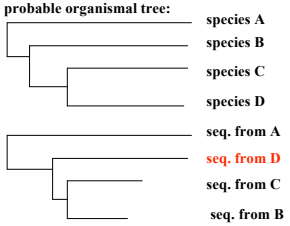
Synologs: genes ended up in one organism through fusion of lineages. The paradigm are genes that were transferred into the eukaryotic cell together with the endosymbionts that evolved into mitochondria and plastids
(the -logs are often spelled with "ue" like in orthologues)

see Fitch's article in TIG 2000 for more discussion.

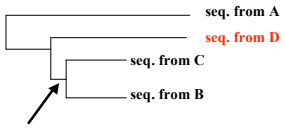
Trees – what might they mean?

Calculating a tree is comparatively easy, figuring out what it might mean is much more difficult.

If this is the probable organismal tree:

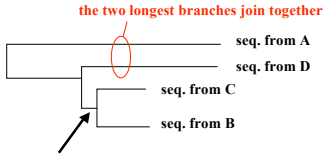


lack of resolution



e.g., 60% bootstrap support for bipartition (AD)(CB)

long branch attraction artifact

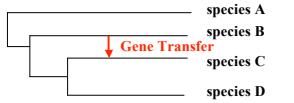


e.g., 100% bootstrap support for bipartition (AD)(CB)

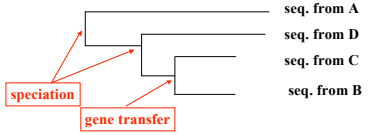
What could you do to investigate if this is a possible explanation?
use only slow positions,
use an algorithm that corrects for ASRV

Gene transfer

Organismal tree:

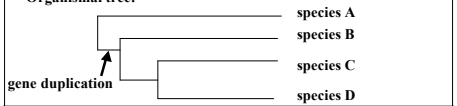


molecular tree:

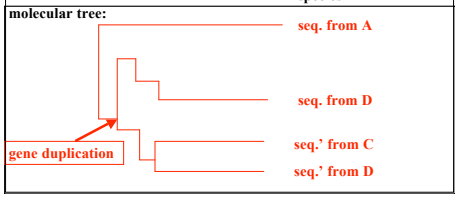


Gene duplication

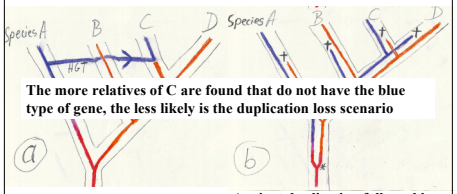
Organismal tree:



molecular tree:



Gene duplication and gene transfer are equivalent explanations.



The more relatives of C are found that do not have the blue type of gene, the less likely is the duplication loss scenario

Horizontal or lateral Gene

Ancient duplication followed by gene loss

Note that scenario B involves many more individual events than A

1 HGT with orthologous replacement

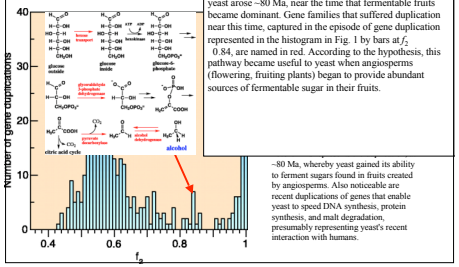
1 gene duplication followed by 4 independent gene loss events

What is it good for?

Gene duplication events can provide an outgroup that allows rooting a molecular phylogeny.
Most famously this principle was applied in case of the tree of life – the only outgroup available in this case are ancient paralogs (see http://gogarten.uconn.edu/cvs/Publ_Pres.htm for more info).
However, the same principle also is applicable to any group of organisms, where a duplication preceded the radiation (example).
Lineage specific duplications also provide insights into which traits were important during evolution of a lineage.

e.g. gene duplications in yeast

from Benner et al., 2002

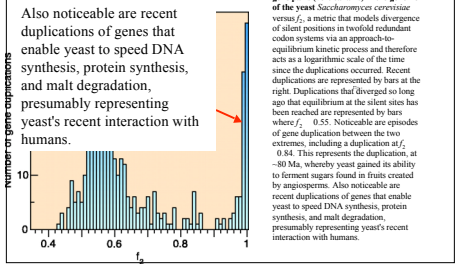


The chemical pathway that converts glucose to alcohol in yeast arose ~80 Ma, near the time that fermentable fruits became dominant. Gene families that suffered duplication near this time, captured in the episode of gene duplication represented in the histogram in Fig. 1 by bars at f_2 0.84, are named in red. According to the hypothesis, this pathway became useful to yeast when angiosperms (flowering, fruiting plants) began to provide abundant sources of fermentable sugar in their fruits.

~80 Ma, whereby yeast gained its ability to ferment sugars found in fruits created by angiosperms. Also noticeable are recent duplications of genes that enable yeast to speed DNA synthesis, protein synthesis, and malt degradation, presumably representing yeast's recent interaction with humans.

e.g. gene duplications in yeast

from Benner et al., 2002

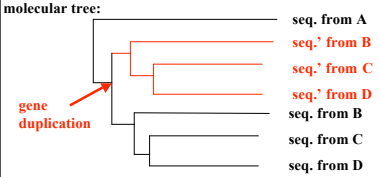


Also noticeable are recent duplications of genes that enable yeast to speed DNA synthesis, protein synthesis, and malt degradation, presumably representing yeast's recent interaction with humans.

Figure 1. The number of duplicated gene pairs (vertical axis) in the genome of the yeast *Saccharomyces cerevisiae* versus f_2, a metric that models divergence of silent positions in twofold redundant codon systems via an approach-to-equilibrium kinetic process and therefore acts as a logarithmic scale of the time since the duplications occurred. Recent duplications are represented by bars at the right. Duplications that diverged so long ago that equilibrium at the silent sites has been reached are represented by bars where f_2 0.55. Noticeable are episodes of gene duplication between the two extremes, including a duplication at f_2 0.84. This represents the duplication, at ~80 Ma, whereby yeast gained its ability to ferment sugars found in fruits created by angiosperms. Also noticeable are recent duplications of genes that enable yeast to speed DNA synthesis, protein synthesis, and malt degradation, presumably representing yeast's recent interaction with humans.

Function, ortho- and paralogy

molecular tree:



The presence of the duplication is a taxonomic character (shared derived character in species B C D).
The phylogeny suggests that seq' and seq have similar function, and that this function was important in the evolution of the clade BCD.
seq' in B and seq'in C and D are orthologs and probably have the same function, whereas seq and seq' in BCD probably have different function (the difference might be in subfunctionalization of functions that seq had in A. – e.g. organ specific expression)

Why phylogenetic reconstruction of molecular evolution?

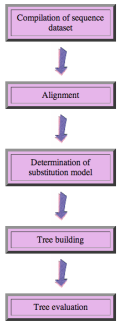
- Systematic classification of organisms. E.g.:
  - Who were the first angiosperms? (i.e. where are the first angiosperms located relative to present day angiosperms?)
  - Where in the tree of life is the last common ancestor located?
- Evolution of molecules. E.g.:
  - domain shuffling,
  - reassignment of function,
  - gene duplications,
  - horizontal gene transfer,
  - drug targets,
  - detection of genes that drive evolution of a species/population (e.g. influenca virus, see here for more examples)

Steps of the phylogenetic analysis

Phylogenetic analysis is an inference of evolutionary relationships between organisms. Phylogenetics tries to answer the question "How did groups of organisms come into existence?"

Those relationships are usually represented by tree-like diagrams.

Note: the assumption of a tree-like process of evolution is controversial!



Phylogenetic reconstruction - How

Distance analyses

calculate pairwise distances
(different distance measures, correction for multiple hits, correction for codon bias)

make distance matrix (table of pairwise corrected distances)

calculate tree from distance matrix

i) using optimality criterion
(e.g.: smallest error between distance matrix and distances in tree, or use
ii) algorithmic approaches (UPGMA or neighbor joining) B)

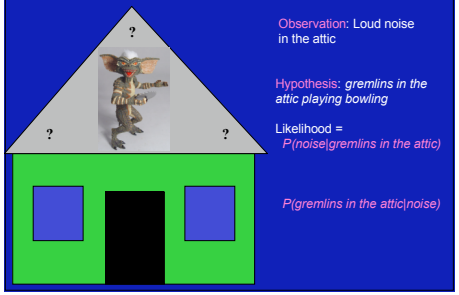
Phylogenetic reconstruction - How

Parsimony analyses
find that tree that explains sequence data with minimum number of substitutions
(tree includes hypothesis of sequence at each of the nodes)

Maximum Likelihood analyses
given a model for sequence evolution, find the tree that has the highest probability under this model.
This approach can also be used to successively refine the model.

Bayesian statistics use ML analyses to calculate posterior probabilities for trees, clades and evolutionary parameters. Especially MCMC approaches have become very popular in the last year, because they allow to estimate evolutionary parameters (e.g., which site in a virus protein is under positive selection), without assuming that one actually knows the "true" phylogeny.

Elliot Sober's Gremlins



Else:
spectral analyses, like evolutionary parsimony, look only at patterns of substitutions,

Another way to categorize methods of phylogenetic reconstruction is to ask if they are using

an optimality criterion (e.g.: smallest error between distance matrix and distances in tree, least number of steps, highest probability), or

algorithmic approaches (UPGMA or neighbor joining)

Packages and programs available: PHYLIP, phyml, MrBayes, Tree-Puzzle, PAUP*, clustalw, raxml, PhyloGenie, PyPhy

Bootstrap ?

- See here

Phylip

written and distributed by Joe Felsenstein and collaborators (some of the following is copied from the PHYLIP homepage)

PHYLIP (the *PHYL*ogeny *I*nference *P*ackage) is a package of programs for inferring phylogenies (evolutionary trees).

PHYLIP is the most widely-distributed phylogeny package, and competes with PAUP* to be the one responsible for the largest number of published trees. PHYLIP has been in distribution since 1980, and has over 15,000 registered users.

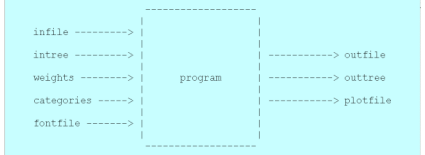
Output is written onto special files with names like "outfile" and "outtree". Trees written onto "outtree" are in the Newick format, an informal standard agreed to in 1986 by authors of a number of major phylogeny packages.

Input is either provided via a file called "infile" or in response to a prompt.

input and output

Input and output files

For most of the PHYLIP programs, information comes from a series of input files, and ends up in a series of output files:



The programs interact with the user by presenting a menu. Aside from the user's choices from the menu, they read all other input from files. These files have default names. The program will try to find a file of that name - if it does not, it will ask the user to supply the name of that file. Input data such as DNA sequences comes from a file whose default name is `infile`. If the user supplies a tree, this is in a file whose default name is `intree`. Values of weights for the characters are in `weights`, and the tree plotting program need some digitized fonts which are supplied in `fontfile` (all these are default names).

What's in PHYLIP

Programs in PHYLIP allow to do parsimony, distance matrix, and likelihood methods, including bootstrapping and consensus trees. Data types that can be handled include molecular sequences, gene frequencies, restriction sites and fragments, distance matrices, and discrete characters.

Phylip works well with protein and nucleotide sequences
Many other programs mimic the style of PHYLIP programs. (e.g. TREEPUZZLE, phyml, protml)

Many other packages use PHYIP programs in their inner workings (e.g., PHYLO_WIN)

PHYLIP runs under all operating systems

Web interfaces are available

Programs in PHYLIP are Modular

For example:

SEQBOOT take one set of aligned sequences and writes out a file containing bootstrap samples.

PROTDIST takes a aligned sequences (one or many sets) and calculates distance matices (one or many)

FITCH (or NEIGHBOR) calculate best fitting or neighbor joining trees from one or many distance matrices

CONSENSE takes many trees and returns a consensus tree

…. modules are available to draw trees as well, but often people use treeview or njplot

The Phylip Manual is an excellent source of information.

Brief one line descriptions of the programs are here

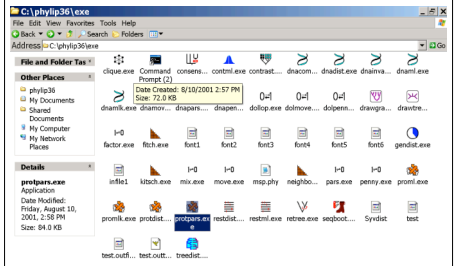
The easiest way to run PHYLIP programs is via a command line menu (similar to clustalw). The program is invoked through clicking on an icon, or by typing the program name at the command line.

```
> seqboot
> protpars
> fitch
```

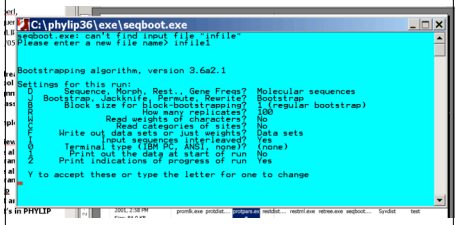
If there is no file called infile the program responds with:

```
[gogarten@carrot gogarten]$ seqboot
seqboot: can't find input file "infile"
Please enter a new file name>
```

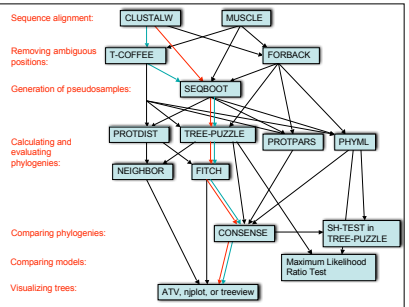
program folder



menu interface



example: seqboot and protpars on infile1



Phylip programs can be combined in many different ways with one another and with programs that use the same file formats.

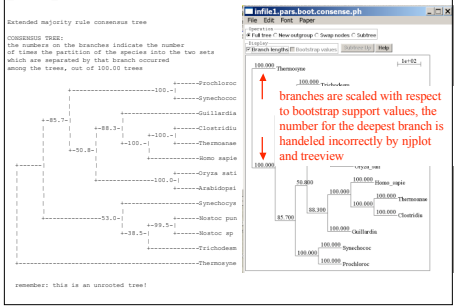
Example 1 Protpars

example: seqboot, protpars, consense on infile1

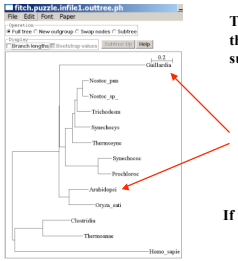
NOTE the bootstrap majority consensus tree does not necessarily have the same topology as the "best tree" from the original data!

threshold parsimony,
gap symbols - versus ?
(in vi you could use `:%s/-/?/g` to replace all – ?)
outfile
outtree compare to distance matrix analysis

protpars (versus distance/FM)



(protpars versus) distance/FM



Tree is scaled with respect to the estimated number of substitutions.

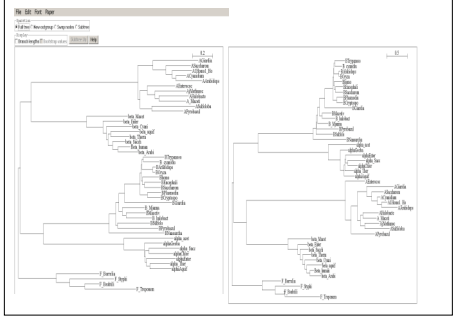
what might be the explanation for the red algae not grouping with the plants?

If time: demo of njplot

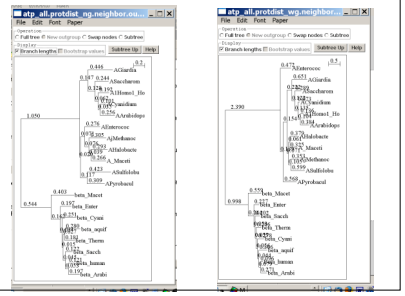
protdist

```
PROTdist
Settings for this run:
 P  Use JTT, PMB, PAM, Kimura, categories model?  Jones-Taylor-Thornton matrix
 G  Gamma distribution of rates among positions?  No
 C        One category of substitution rates?  Yes
 W           Use weights for positions?  No
 M           Analyze multiple data sets?  No
 I        Input sequences interleaved?  Yes
 0        Terminal type (IBM PC, ANSI)?  ANSI
 1        Print out the data at start of run  No
 2        Print indications of progress of run  Yes
```

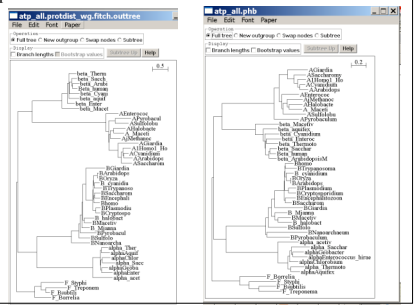
without and with correction for ASRV



subtree with branch lengths
without and with correction for ASRV

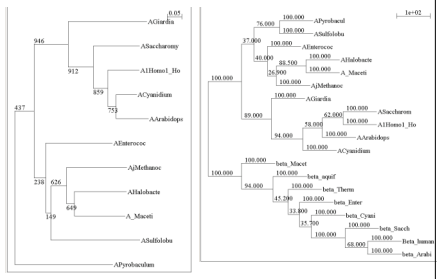


compare to trees with FITCH and clustalw – same dataset



bootstrap support ala clustal

protpars (gaps as ?)



phyml

PHYML - A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood

An online interface is here ;
there is a command line version that is described here (not as straight forward as in clustalw);
a phylip like interface is automatically invoked, if you type "phyml" – the manual is here.

Phyml is installed on bbcxsrv1.

Do example on atp_all.phy
Note data type, bootstrap option within program, models for ASRV (pinvar and gamma), by default the starting tree is calculated via neighbor joining.

phyml - comments

Under some circumstances the consensus tree calculated by phyml is wrong. It is recommended to save all the individual trees and to also evaluate them with *consense* from the phylip package.
Note: phyml allows longer names, but consense allows only 10 characters!

phyml is fast enough to analyze dataset with hundreds of sequences (in 1990, a maximum likelihood analyses with 12 sequences (no ASRV) took several days).

For moderately sized datasets you can estimate branch support through a bootstrap analysis (it still might run several hours, but compared to protml or PAUP, this is extremely fast).

The paper describing phyml is here,
a brief interview with the authors is here

TreePuzzle ne PUZZLE

TREE-PUZZLE is a very versatile maximum likelihood program that is particularly useful to analyze protein sequences. The program was developed by Korbian Strimmer and Arnd von Haseler (then at the Univ. of Munich) and is maintained by von Haseler, Heiko A. Schmidt, and Martin Vingron

(contacts see http://www.tree-puzzle.de/).

TREE-PUZZLE

- allows fast and accurate estimation of ASRV (through estimating the shape parameter alpha) for both nucleotide and amino acid sequences,
- It has a "fast" algorithm to calculate trees through quartet puzzling (calculating ml trees for quartets of species and building the multispecies tree from the quartets).
- The program provides confidence numbers (puzzle support values), which tend to be smaller than bootstrap values (i.e. provide a more conservative estimate),
- the program calculates branch lengths and likelihood for user defined trees, which is great if you want to compare different tree topologies, or different models using the **maximum likelihood ratio test**.
- Branches which are not significantly supported are collapsed.
- TREE-PUZZLE runs on "all" platforms
- TREE-PUZZLE reads PHYLIP format, and communicates with the user in a way similar to the PHYLIP programs.

Maximum likelihood ratio test

If you want to compare two models of evolution (this includes the tree) given a data set, you can utilize the so-called maximum likelihood ratio test.
If L_1 and L_2 are the likelihoods of the two models, $d = 2(\log L_1 - \log L_2)$ approximately follows a Chi square distribution with n degrees of freedom. Usually n is the difference in model parameters. I.e., how many parameters are used to describe the substitution process and the tree. In particular n can be the difference in branches between two trees (one tree is more resolved than the other).
In principle, this test can only be applied if on model is a more refined version of the other. In the particular case, when you compare two trees, one calculated without assuming a clock, the other assuming a clock, the degrees of freedom are the number of OTUs – 2 (as all sequences end up in the present at the same level, their branches cannot be freely chosen) .

To calculate the probability you can use the CHISQUARE calculator for windows available from Paul Lewis.

TREE-PUZZLE allows (cont)

- TREEPUZZLE calculates distance matrices using the ml specified model. These can be used in FITCH or Neighbor.
PUZZLEBOOT automates this approach to do bootstrap analyses – WARNING: this is a distance matrix analyses!
The official script for PUZZLEBOOT is here – you need to create a command file (puzzle.cmds), and puzzle needs to be envocable through the command puzzle.
Your input file needs to be the renamed outfile from **seqboot**
A slightly modified working version of puzzleboot_mod.sh is here, and here is an example for puzzle.cmds . Read the instructions before you run this!
- Maximum likelihood mapping is an excellent way to assess the phylogenetic information contained in a dataset.
- ML mapping can be used to calculate the support around one branch.

@@@ Puzzle is cool, don't leave home without it! @@@

Bayes' Theorem

Reverend Thomas Bayes (1702-1761)

$$P(model|data, I) = P(model, I)\,\frac{P(data|model, I)}{P(data, I)}$$

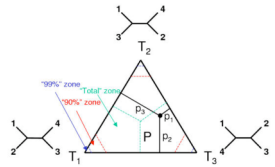
Likelihood: describes how well the model predicts the data

Posterior Probability: represents the degree to which we believe a given **model** accurately describes the situation given the available **data** and all of our prior information I

Prior Probability: describes the degree to which we believe the model accurately describes reality based on all of our prior information.

Normalizing constant

ml mapping



From: Olga Zhaxybayeva and J Peter Gogarten *BMC Genomics* 2002, 3:4

ml mapping

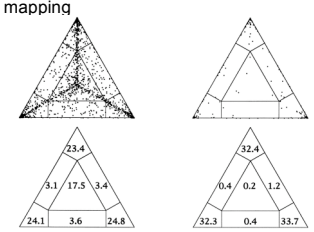
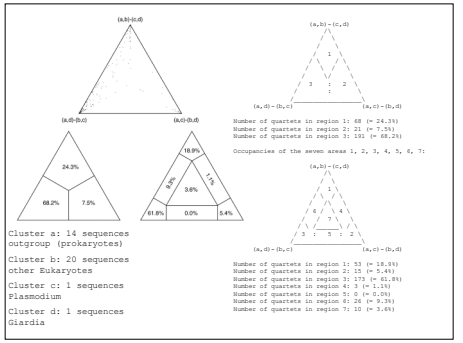


Figure 5. Likelihood-mapping analysis for two biological data sets. (*Upper*) The distribution patterns. (*Lower*) The occupancies (in percent) for the seven areas of attraction. (*A*) Cytochrome-*b* data from ref. 14. (*B*) Ribosomal DNA of major arthropod groups (15).

From: Korbinian Strimmer and Arndt von Haeseler Proc. Natl. Acad. Sci. USA Vol. 94, pp. 6815-6819, June 1997



Alternative Approaches to Estimate Posterior Probabilities

Bayesian Posterior Probability Mapping with MrBayes
(Huelsenbeck and Ronquist, 2001)

Problem:

Strimmer's formula $p_i = \frac{L_i}{L_1+L_2+L_3}$ only considers 3 trees (those that maximize the likelihood for the three topologies)

Solution:

Exploration of the tree space by sampling trees using a biased random walk (Implemented in MrBayes program)

Trees with higher likelihoods will be sampled more often

$p_i \approx \frac{N_i}{N_{total}}$,where N_i - number of sampled trees of topology i, i=1,2,3

N_{total} – total number of sampled trees (has to be large)

Illustration of a biased random walk

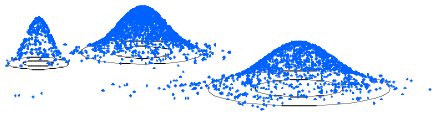


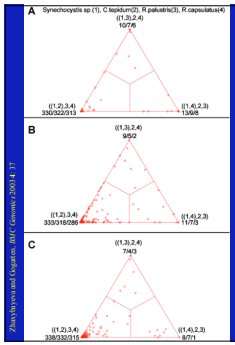
Figure generated using MCRobot program (Paul Lewis, 2001)

ml mapping (cont)

If we want to know if *Giardia lamblia* forms the deepest branch within the known eukaryotes, we can use ML mapping to address this problem.
To apply ml mapping we choose the "higher" eukaryotes as cluster a, another deep branching eukaryote (the one that competes against Giardia) as cluster b, Giardia as cluster c, and the outgroup as cluster d. For an example output see this sample ml-map.

An analysis of the carbamoyl phosphate synthetase domains with respect to the root of the tree of life is here.

Application of ML mapping to comparative Genome analyses
see here for a comparison of different probabil;ity measures
see here for an approach that solves the problem of poor taxon sampling that is usually considered inherent with quartet analyses is.

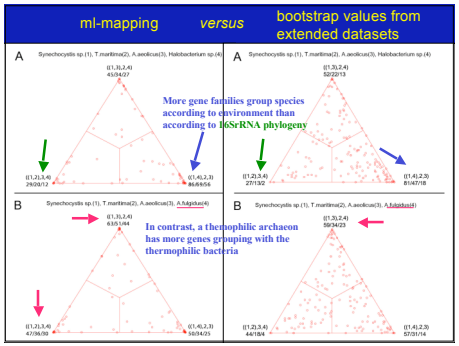


COMPARISON OF DIFFERENT SUPPORT MEASURES

A: mapping of posterior probabilities according to Strimmer and von Haeseler

B: mapping of bootstrap support values

C: mapping of bootstrap support values from extended datasets



TREE-PUZZLE – PROBLEMS/DRAWBACKS

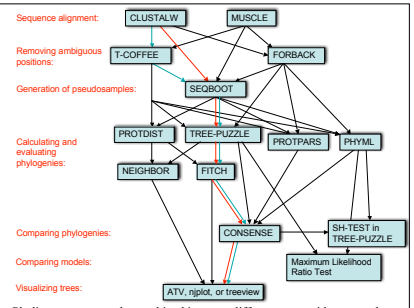
- The more species you add the lower the support for individual branches. While this is true for all algorithms, in TREE-PUZZLE this can lead to completely unresolved trees with only a few handful of sequences.
- Trees calculated via quartet puzzling are usually not completely resolved, and they do not correspond to the ML-tree: The determined multi-species tree is not the tree with the highest likelihood, rather it is the tree whose topology is supported through ml-quartets, and the lengths of the resolved branches is determined through maximum likelihood.

puzzle example

archaea_euk.phy in puzzle_temp

usertree

check outfile



Phylip programs can be combined in many different ways with one another and with programs that use the same file formats.

Old Assignments

Read chapter 4 in Learning Perl

Turn your script that calculates the reverse complement of a sequence into a subroutine

Write a script that takes all files with the extension .fa (containing a single fasta formated sequence) and writes their contents in a single multiple sequence file.

Rev_comp; solution #1

```
#!/usr/bin/perl -w
##############INPUT====================
#input sequence; chomp every line, and concatenate into one big scalar called $seq
    unless($#ARGV==1) {die "please provide name of the file in the command line!!\n";}
    $filename=$ARGV[0];
    open(IN, "< $filename") or die "cannot open $filename:$!";

    $seq='';
    while(defined($line=<IN>)){

            chomp($line);
            $seq .= $line ;
    }
# print "input sequence is  \n $seq\n"; #make sure sequence is read accurately
########## END INPUT#########################

####Program: call sub and print output

$invcomp=&RevComp($seq) ;#call subroutine, hand over content of $seq as parameter
# the result of the suroutines operation will be assigned to $invcomp

print "\n\n\ninverse complement is \n$invcomp\n"; #print output

########End Program###############
#### Subs:
sub RevComp {
    my  ($DNAseq, $rev, $rev_comp);# 'local' variables
#   print "subroutine RevComp was called\n"; #make sure sub is called
    $DNAseq=$_[0];
#   print "subroutine is working on \n $DNAseq\n";# make sure sub got the right sequence
    $rev = reverse $DNAseq;
        $rev_comp=$rev;
    $rev_comp =~ tr/atgcATGC/TACGTACG/;
        $rev_comp; # returns contents of this variable to the main program - sometimes superfluous-
        # NOT here, without it the subroutine returns the number of translations made
    };
```

Rev_comp; solution #2

```
#!/usr/bin/perl -s
#Dustin's version
my $seq='';
my $revcomp='';
PRINT: {
    print "Please enter sequence to be reverse complemented\n";
    chomp($seq=<>);
    $revcomp=Reverse(Complement($seq));
    &Complement();
    print "\n";
    print "The reverse complement of \$seq=$seq" is"." $revcomp\n\n";
exit;}

sub Reverse {
    my $string = $_[0];
    $rev= reverse $string;
    return $rev;
}

sub Complement{
    my$comp= $_[0];
    $comp =~ tr/ATGCVRatogyr/TACGBYTACGBYV/;
    return $comp;
}
```

Old Assignments

Write a script that takes all files with the extension .fa (containing a single fasta formated sequence) and writes their contents in a single multiple sequence file.

Simple solution:
From the shell

```
cat *.fa > all.faa
```

From within a script

```
...
system ("cat *.fa >> all.faa");
```

Old Assignments

complex solution:

```
#!usr/bin/perl/ -w
$outname="";# reset name of outfile
open (OUT, ">temp.out") || die "cannot open: $!";#open a file for output -
# as the names of the infiles have not been read yet, I assign this a temporary name to be renamed later
#
while(defined($file=glob("*.fa"))){
#loop opens one file ending in .fa after the other. stores the name in $file
    open (IN, "<$file") || die "cannot open: $!";
    #reports error if file cannot be opened
    #and assigns $file to IN
    @filename_parts=split(/\./,$file);
    # writes first part of filename into $filename_parts[0]
        $outname .= "$filename_parts[0].";
        ### concatenate $filename_part[0] into new outname ###
    #repeats this for every cycle
    while (defined($line=<IN>)){print OUT $line;}
    #writes contents of $file line by line into filehandle OUT
} #no more *.fa files in directory
close OUT; #closes filehandle OUT and temp.out
$outname="$outname".'faa';
#adds useful extension to outfile
system ("mv temp.out $outname");
# uses unix command mv (move)to rename temp.out into new $outname
```

old assignment 2

Assume that you have the following non-aligned multiple sequence files in a directory:

A.fa : vacuolar/archaeal ATPase catalytic subunits ;
B.fa : vacuolar/archaeal ATPase non-catalytic subunits;
alpha.fa : F-ATPases non-catalytic subunits,
beta.fa : F-ATPases catalytic subunits,
F.fa : ATPase involved in the assembly of the bacterial flagella.

Write a perl script that executes muscle or clustalw and
1) aligns the sequences within each file
2) successively calculates profile alignments between all aligned sequences.

Hints:
`system (command) ;` # executes "command"as if you had typed `command` in the command line

Something like this….

```
#!/usr/bin/perl
#assumption: all files we want to align have ".fa" extension


@files=glob("*.fa"); #in this context returns all file names
$num_files=@files;

$counter=0;
while(defined($file=glob("*.fa"))){
    @filename_parts=split(/\./,$file);
    $aln_file=$filename_parts[0].".aln";
    if(-e $aln_file){ #-e is a file test operator, asks does file exist
    print "$file was already aligned\n";
    }
    else{
    system("clustalw -align -infile=$file -type=protein");
    $counter++;
    }
}
print "$counter files out of $num_files were aligned\n";
```

Or a more pedestrian approach:

```
#!/usr/bin/perl


system (muscle -in VatpA.fa -out VatpA.afa);
system (muscle -in VatpA.afa -out VatpA.rafa -refine);
system (muscle -in beta.fa -out beta.afa);
system (muscle -in beta.afa -out beta.rafa -refine);
system (muscle -profile -in1 beta.rafa -in2 VatpA.rafa -out Abeta.afa);
system (muscle -refine -in Abeta.afa -out Abeta.rafa);

system (muscle -in VatpA.fa -out VatpA.afa);
system (muscle -in VatpA.afa -out VatpA.rafa -refine);
system (muscle -in beta.fa -out beta.afa);
system (muscle -in beta.afa -out beta.rafa -refine);
system (muscle -profile -in1 beta.rafa -in2 VatpA.rafa -out Abeta.afa);
system (muscle -refine -in Abeta.afa -out Abeta.rafa);

  ......
```

Challenge (postponed):

Often one wants to build families of homologous proteins extracted from genomes. One way to do so is to find reciprocal best hits.
Tools:
The script *blastall.pl* takes the genomes indicted in the first line and calculates all possible genome against genome searches.

This script *simple_rbh_pairs.pl* takes two blastall searches (genome A versus genome B) in -m8 format and listing only the top scoring blast hit for each query) and writes the GI numbers of reciprocal best hits into a table.

The script *run_pairs.pl* runs all possible pairwise extractions of RBHs

Task: write a script that combines the pairwise tables keeping only those families that have a strict reciprocal best blast hit relationship in all genomes.

Perl assignment

Write a script that takes all phylip formated aligned multiple sequence files present in a directory, and performes a bootstrap analyses using maximum parsimony.

Files you might want to use are A.fa, B.fa, alpha.fa, beta.fa, and atp_all.phy. BUT you first have to convert them to phylip format AND you should replace some or all gaps with ?
(In the end you would be able to answer the question "does the resolution increase if a more related subgroup is analyzed independent from an outgroup?)

hints

Rather than typing commands at the menu, you can write the responses that you would need to give via the keyboard into a file (e.g. your_input.txt)

You could start and execute the program protpars by typing

protpars < your_input.txt

your input.txt might contain the following lines:

```
infile1.txt
r
t
10
y
r
r
```

in the script you could use the line

```
system ("protpars < your_input.txt");
```

The main problem are the owerwrite commands if the oufile and outtree files are already existing. You can either create these beforehand, or erase them by moving (mv) their contents somewhere else.

create *.phy files

the easiest (probably) is to run clustalw with the phylip option:
For example (here):

```
#!/usr/bin/perl -w
print "# This program aligns all multiple sequence files with names *.fa \n
# found in its directory using clustalw, and saves them in phyip format.\n";
while(defined($file=glob("*.fa"))){
        @parts=split(/\./,$file);
        $file=$parts[0];
        system("clustalw -infile=$file.fa -align -output=PHYLIP");
        } ;
# cleanup:
system ("rm *.dnd");
exit;
```

Alternatively, you could use a web version of readseq – this one worked great for me ☺

Alternative for entering the commands for the menu:

```
#!/usr/bin/perl -w
        system ("cp A.phy infile");
        system ("echo -e 'y\n9\n'|seqboot");
exit;
```

`echo` **returns the string in** `' ', i.e., y\n9\n.`
The `-e` **options allows the use of** `\n`
The `|` **symbol pipes the output from echo to seqboot**

Other New Assignments:

- **Read chapters 5 and 6**
- **Write a script that determines the number of elements in a %ash.**
- **Write a script (or subroutine) that prints out a hash sorted on the keys in alphabetical order.**
- **How can you remove an entry in a hash (key and value)?**
- **Write a program that it uses hashes to calculates mono-, di-, tri-, and quartet-nucleotide frequencies.**

ml mapping can asses the topology surrounding an individual branch :

E.g.: If we want to know if *Giardia lamblia* forms the deepest branch within the known eukaryotes, we can use ML mapping to address this problem.
To apply ml mapping we choose the "higher" eukaryotes as cluster a, another deep branching eukaryote (the one that competes against Giardia) as cluster b, Giardia as cluster c, and the outgroup as cluster d. For an example output see this sample ml-map.

An analysis of the carbamoyl phosphate synthetase domains with respect to the root of the tree of life is here.

ml mapping can assess the not necessarily treelike histories of genome

Application of ML mapping to comparative Genome analyses

see here for a comparison of different probability measures.
Fig. 3: outline of approach
Fig. 4: Example and comparison of different measures

see here for an approach that solves the problem of poor taxon sampling that is usually considered inherent with quartet analyses.
Fig. 2: The principle of "analyzing extended datasets to obtain embedded quartets"
Example next slides: