MCB 5472

Phylogenetic reconstruction #2

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

OldAssignments:

- **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 in a genome.**

exercises:

Write a script that determines the number of elements in %ash.

```
@keys = keys(%ash); #assigns keys to an array
$number =@keys; # determines number of different keys (uses array in scalar context).
print "$number \n";
```

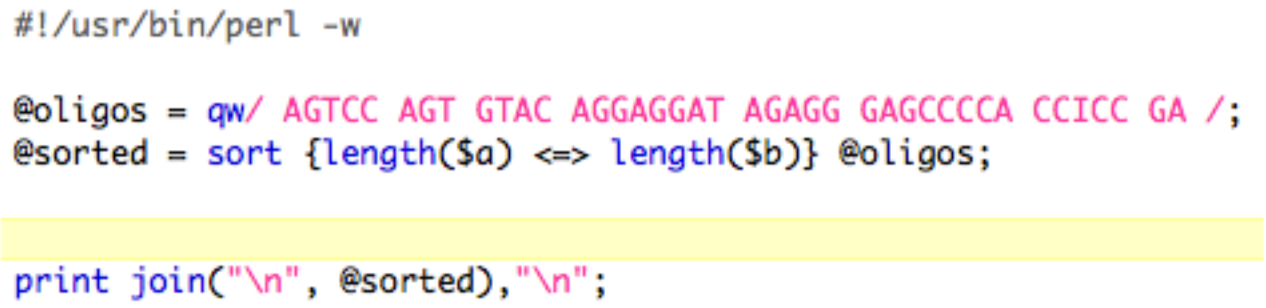
Write a script that prints out a hash sorted on the keys in alphabetical order.

```
@gi_names = sort(keys(%gi_hash)); # sorts key and assigns keys to an array
foreach (@gi_names){
    print "$_ occurred $gi_hash{$_} times\n";
    }
```

Remove an entry in a hash (key and value):

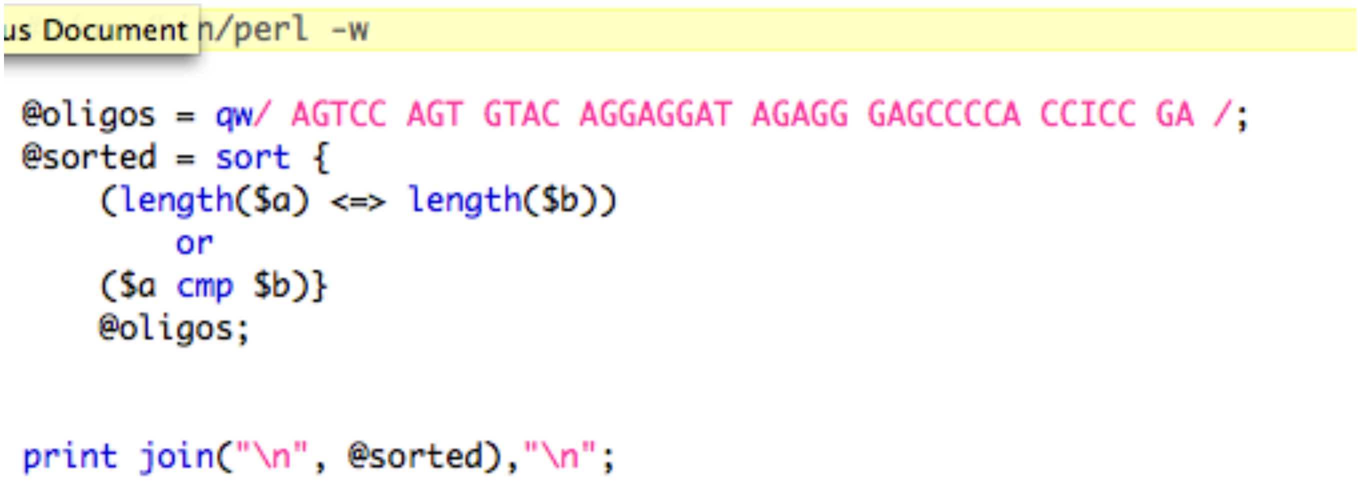
```
delete $gi_hash{$varaible_denoting_some_key};
```

Sort based on code {}



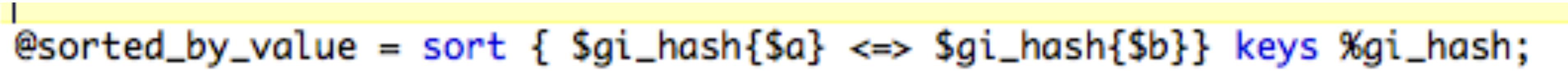
```
#!/usr/bin/perl -w

@oligos = qw/ AGTCC AGT GTAC AGGAGGAT AGAGG GAGCCCCA CCICC GA /;
@sorted = sort {length($a) <=> length($b)} @oligos;


print join("\n", @sorted),"\n";
```

or



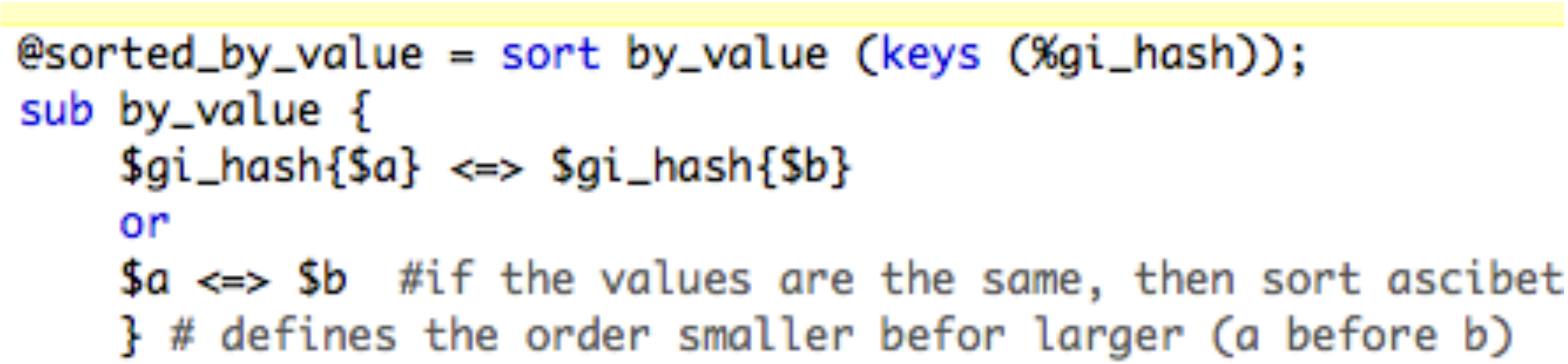
```
us Document n/perl -w

@oligos = qw/ AGTCC AGT GTAC AGGAGGAT AGAGG GAGCCCCA CCICC GA /;
@sorted = sort {
    (length($a) <=> length($b))
        or
    ($a cmp $b)}
    @oligos;


print join("\n", @sorted),"\n";
```

To sort Keys by their value (See class 4 &5):



```
@sorted_by_value = sort { $gi_hash{$a} <=> $gi_hash{$b}} keys %gi_hash;
```

or



```
@sorted_by_value = sort by_value (keys (%gi_hash));
sub by_value {
    $gi_hash{$a} <=> $gi_hash{$b}
    or
    $a <=> $b  #if the values are the same, then sort ascibet
    } # defines the order smaller befor larger (a before b)
```

Old Perl assignments

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

Very Old Assignment

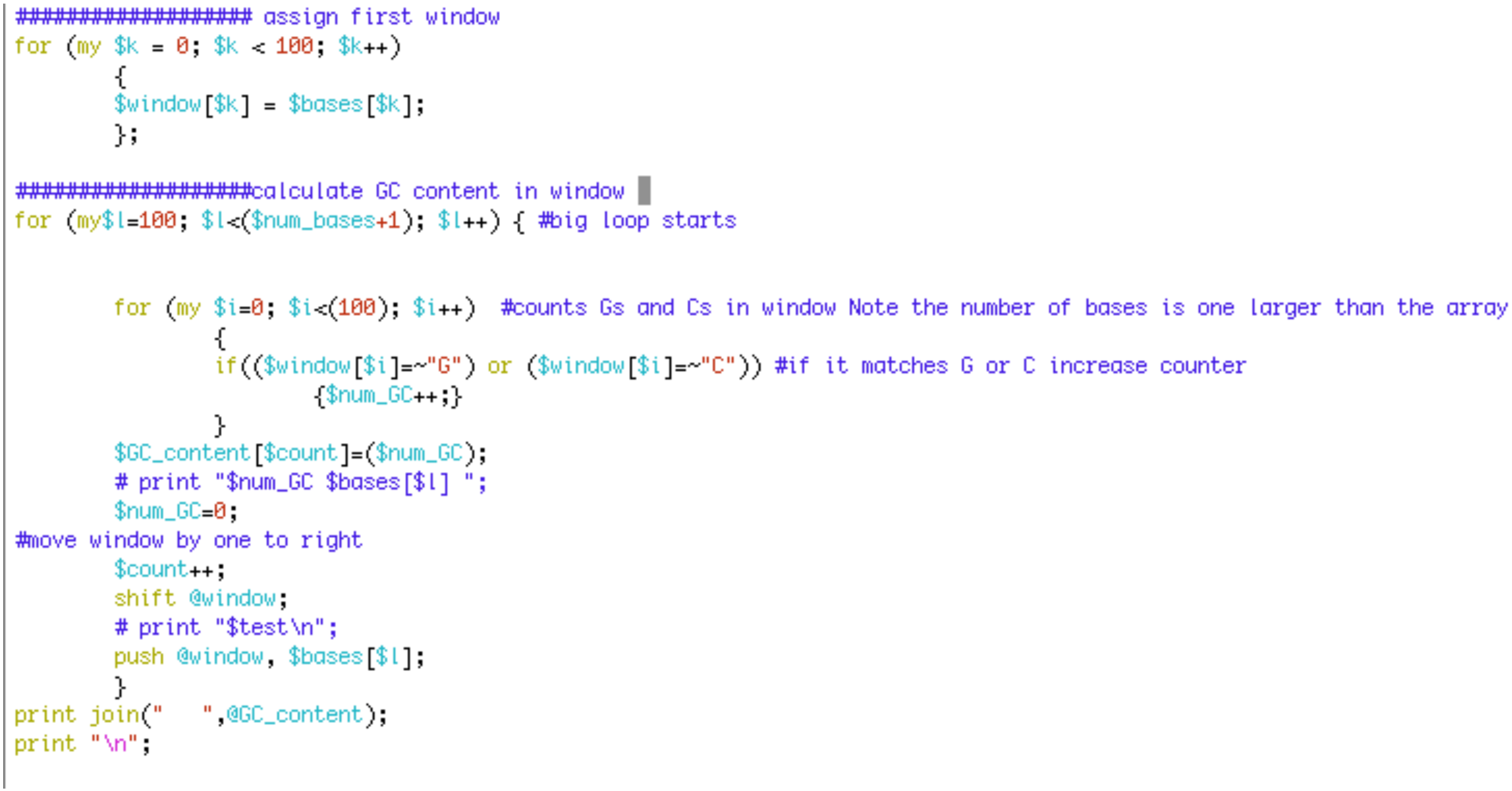
Write a script that takes a genome (fna file) and calculates the GC content.

Modify the script to count tetra- and penta-nucleotide frequencies.

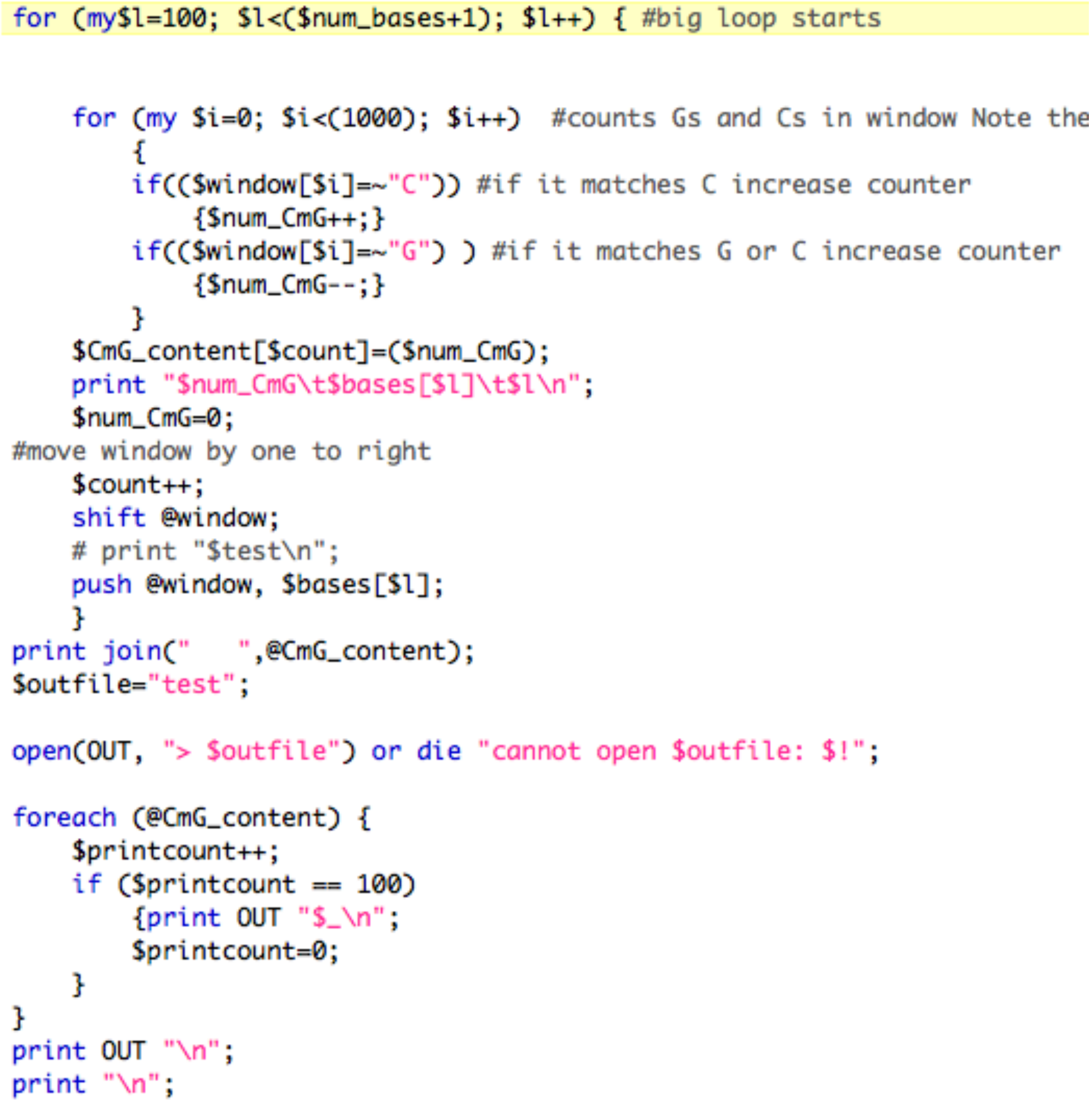
Modify the script so that it creates a table that gives the GC, tetra- and penta-nucleotide content in a sliding window moving through the genome.

(For which of these programs, and for which problems might you want to consider, or correct for strand bias?)

Rolling window example from GC rolling.pl

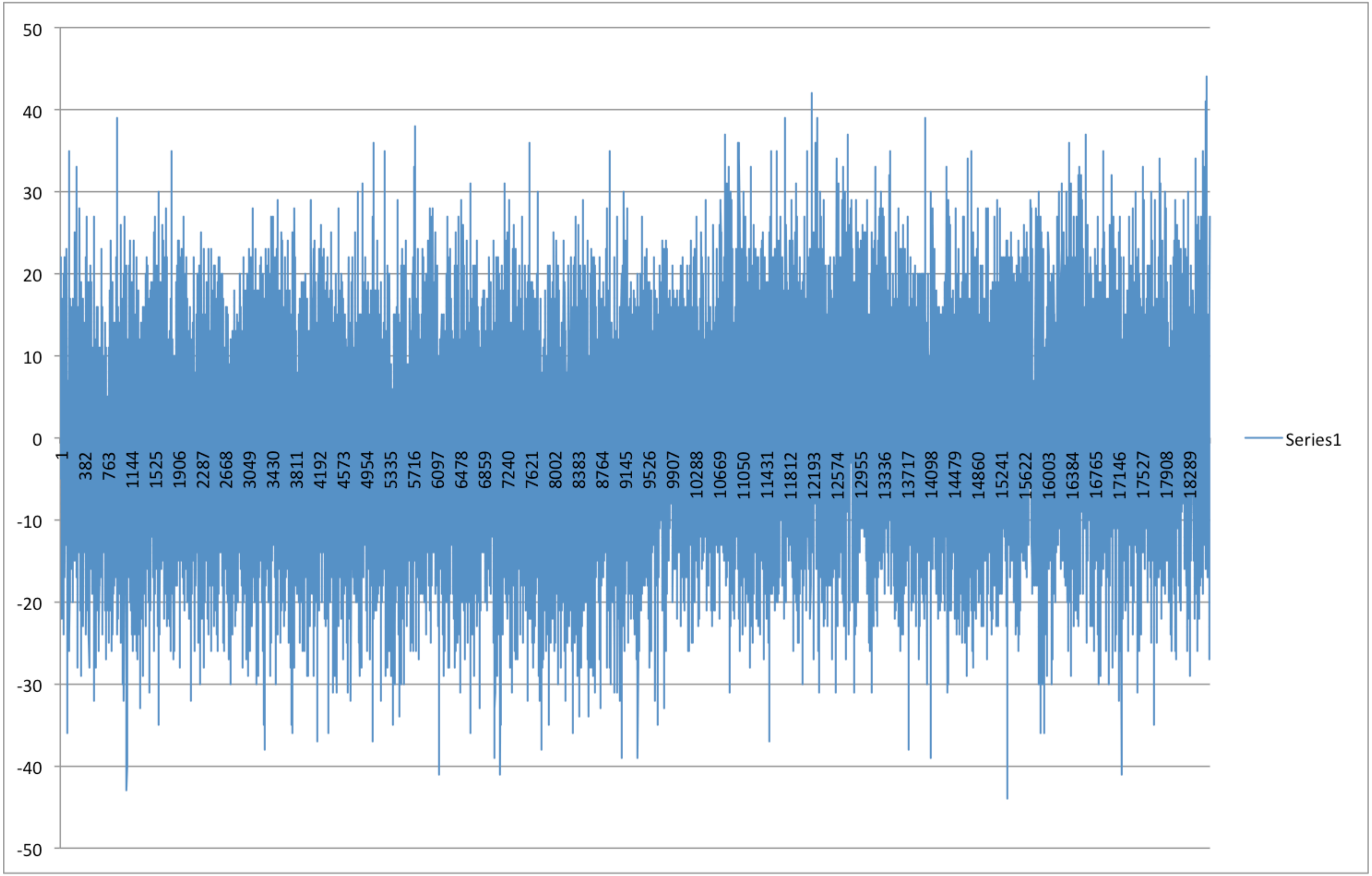


```
################### assign first window
for (my $k = 0; $k < 100; $k++)
        {
        $window[$k] = $bases[$k];
        };

###################calculate GC content in window
for (my$l=100; $l<($num_bases+1); $l++) { #big loop starts


        for (my $i=0; $i<(100); $i++)  #counts Gs and Cs in window Note the number of bases is one larger than the array
                {
                if(($window[$i]=~"G") or ($window[$i]=~"C")) #if it matches G or C increase counter
                        {$num_GC++;}
                }
        $GC_content[$count]=($num_GC);
        # print "$num_GC $bases[$l] ";
        $num_GC=0;
#move window by one to right
        $count++;
        shift @window;
        # print "$test\n";
        push @window, $bases[$l];
        }
print join("   ",@GC_content);
print "\n";
```



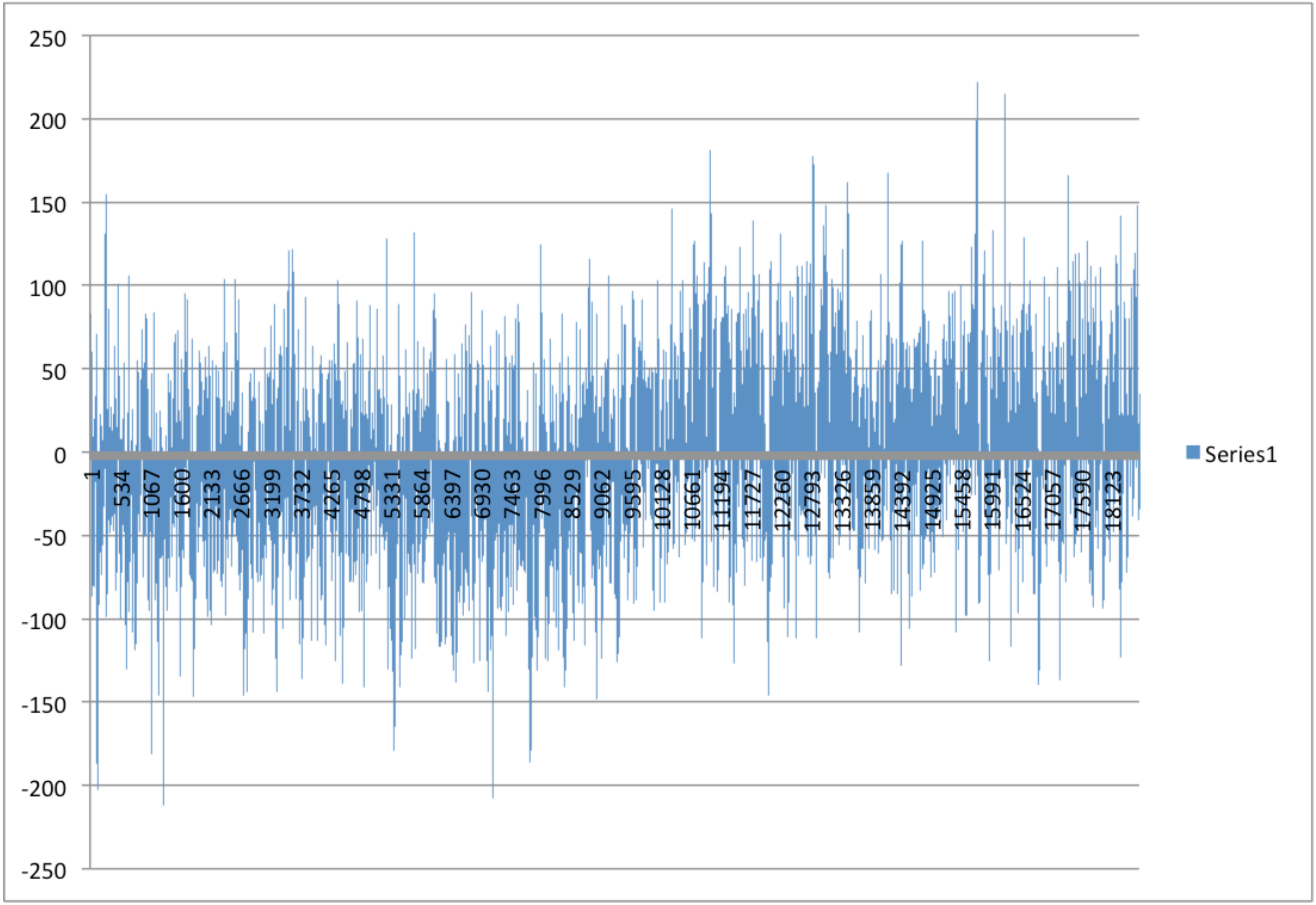
```
for (my$l=100; $l<($num_bases+1); $l++) { #big loop starts


    for (my $i=0; $i<(1000); $i++)  #counts Gs and Cs in window Note the
        {
        if(($window[$i]=~"C")) #if it matches C increase counter
            {$num_CmG++;}
        if(($window[$i]=~"G") ) #if it matches G or C increase counter
            {$num_CmG--;}
        }
    $CmG_content[$count]=($num_CmG);
    print "$num_CmG\t$bases[$l]\t$l\n";
    $num_CmG=0;
#move window by one to right
    $count++;
    shift @window;
    # print "$test\n";
    push @window, $bases[$l];
    }
print join("   ",@CmG_content);
$outfile="test";

open(OUT, "> $outfile") or die "cannot open $outfile: $!";

foreach (@CmG_content) {
    $printcount++;
    if ($printcount == 100)
        {print OUT "$_\n";
        $printcount=0;
    }
}
print OUT "\n";
print "\n";
```

Thermus thermophilus SG0.5JP17-16



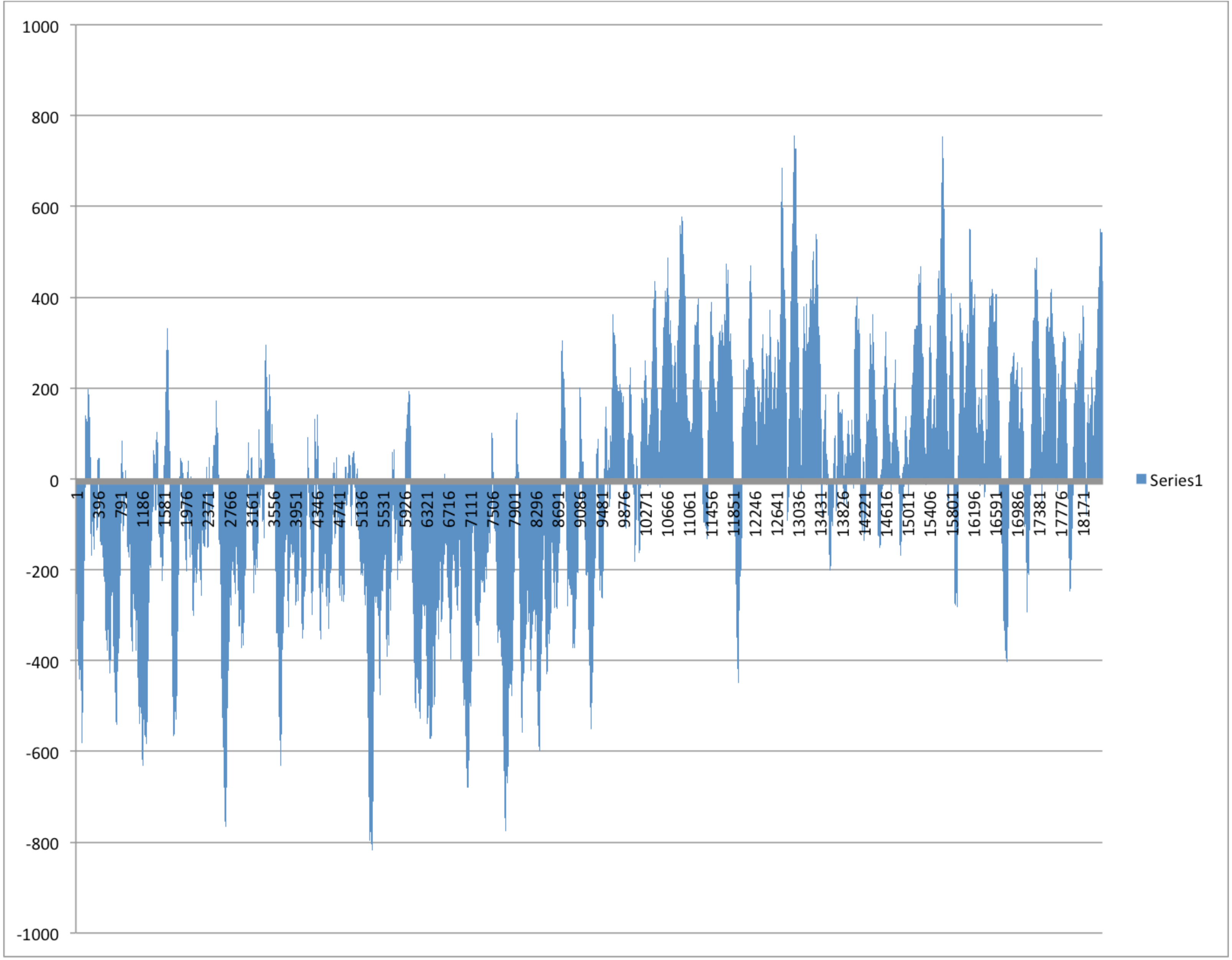
Window=100 , printed every 100

Thermus thermophilus SG0.5JP17-16



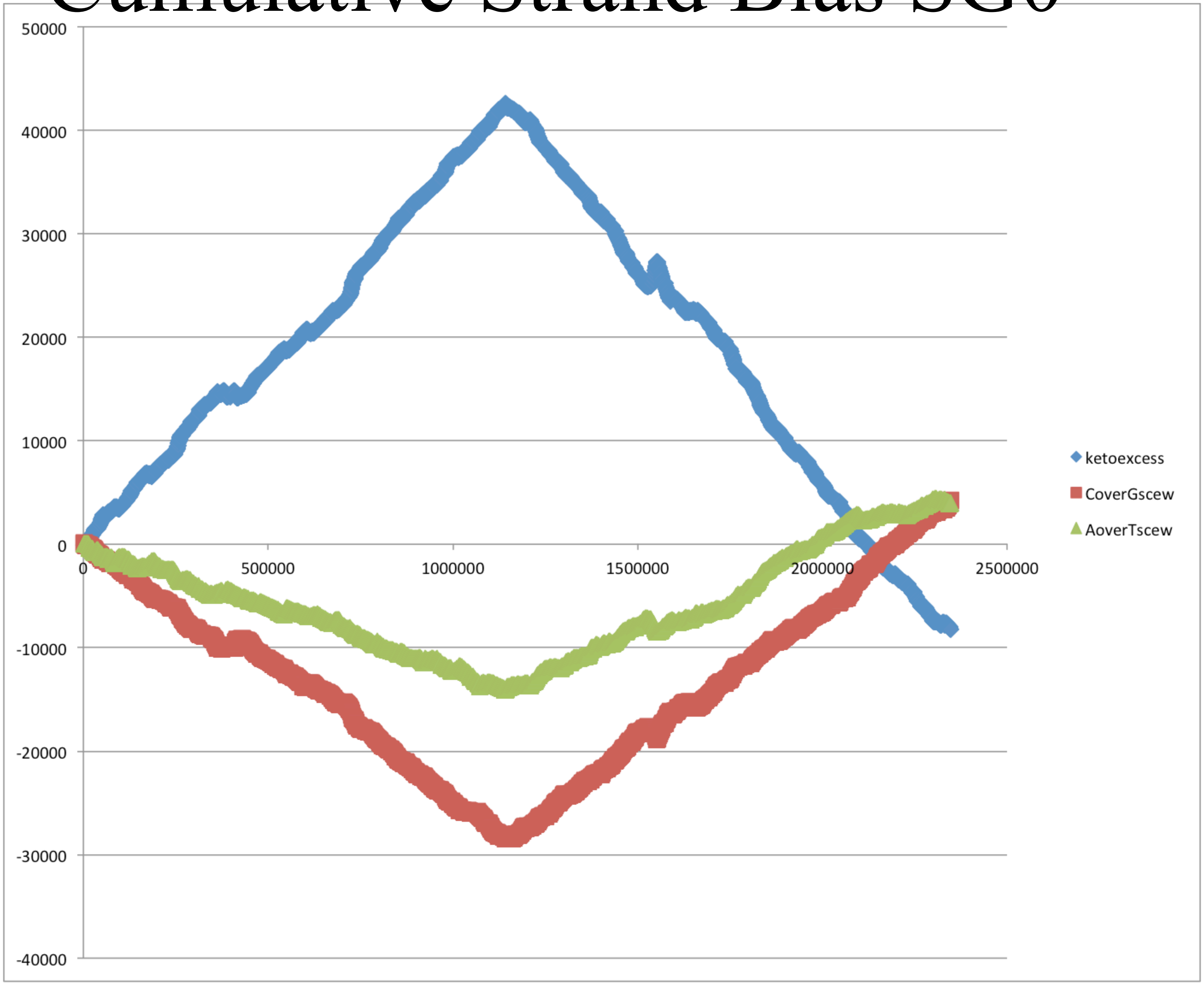
Window=1000 , printed every 100

Thermus thermophilus SG0.5JP17-16



Window=10000 , printed every 100

Cumulative Strand Bias SG0



New Assignment

•Given a multiple fasta sequence file*, write a script that for each sequence extract the gi number and the species name. and rewrites the file so that the annotation line starts with the gi number, followed by the species/strain name, followed by a space. (The gi number and the species name should not be separated by or contain any spaces – replace them by _. This is useful, because clustalw will recognize the number and name as handle for the sequence.)

Assume that the annotation line follows the NCBI convention and begins with the > followed by the gi number, and ends with the species and strain designation given in []

Example:

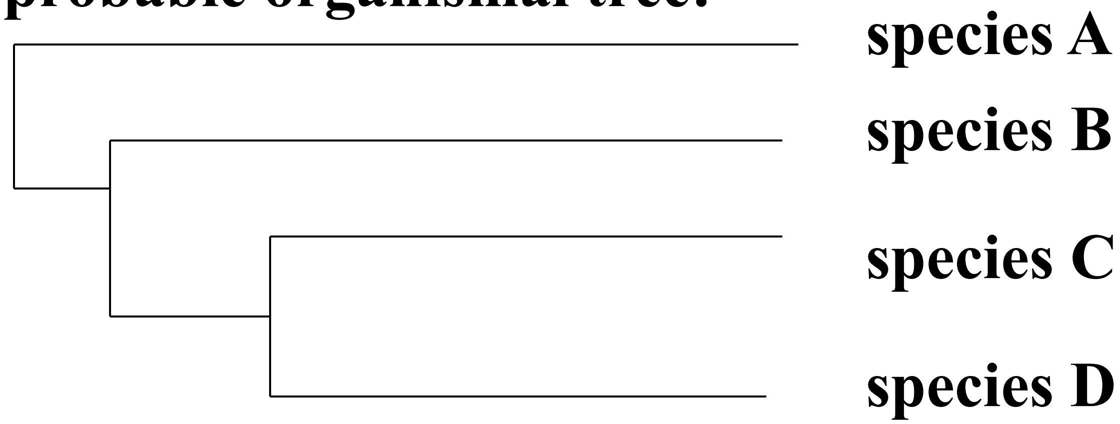
```
>gi|229240723|ref|ZP_04365119.1| primary replicative DNA
helicase; intein [Cellulomonas flavigena DSM 20109]
```

Example multiple sequence file is here.

- Work on your student project

Trees – possible reasons for conflict

If this is the probable organismal tree:



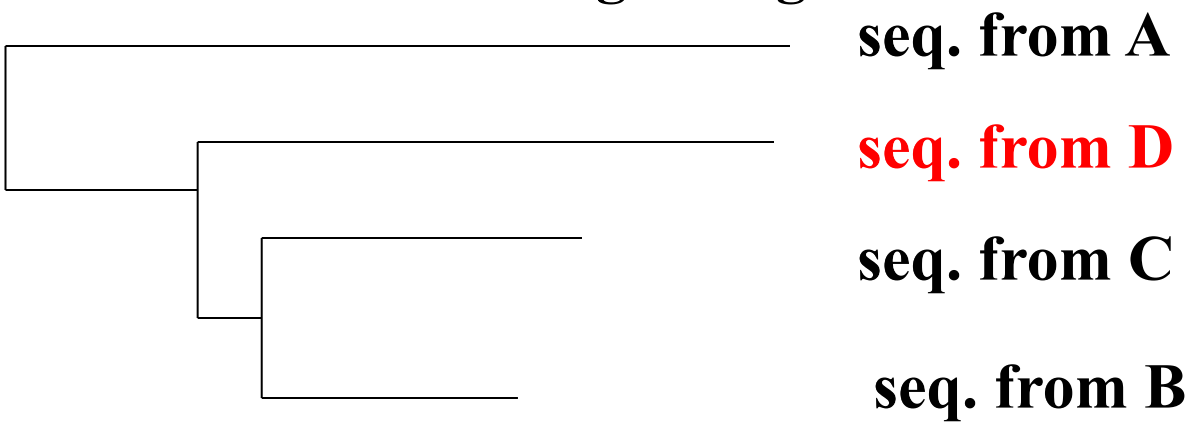
species A

species B

species C

species D

what could be the reason for obtaining this gene tree:



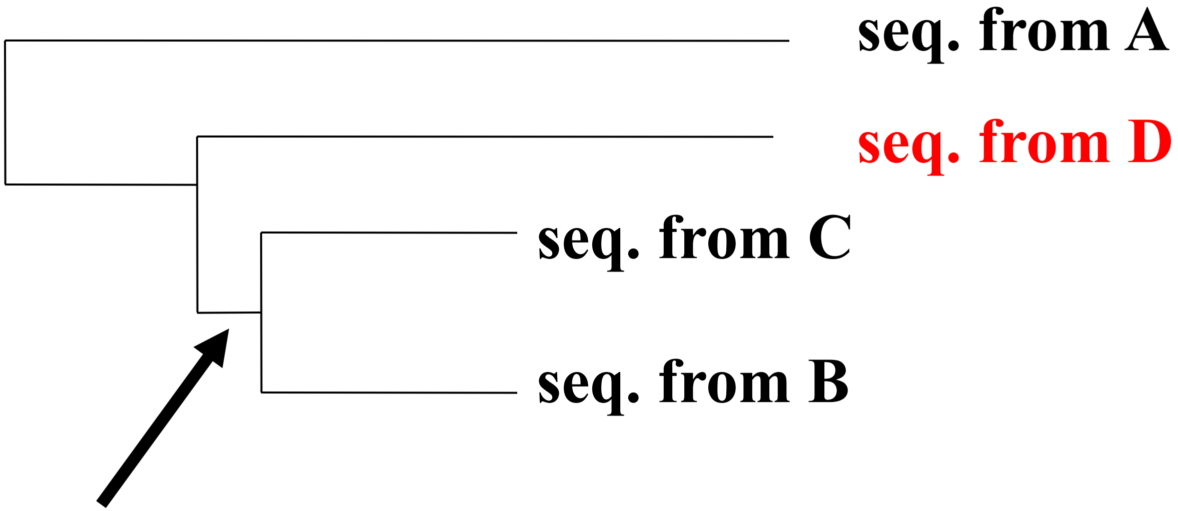
seq. from A

seq. from D

seq. from C

seq. from B

lack of resolution



seq. from A

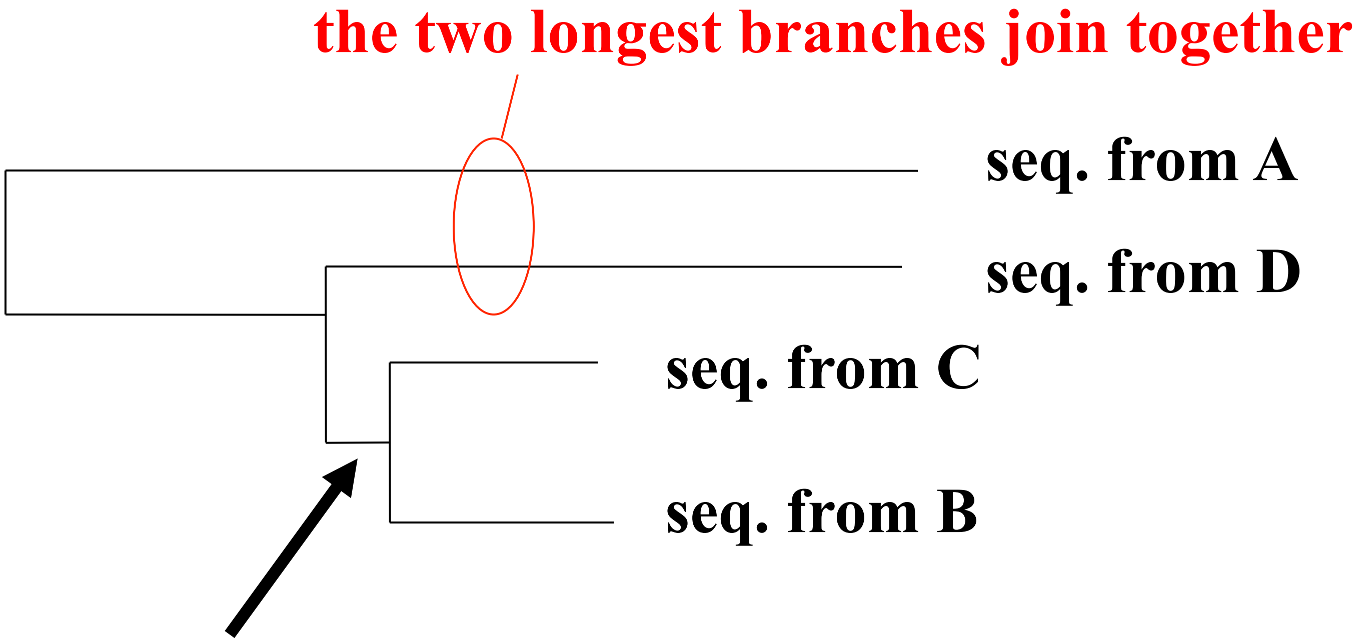
seq. from D

seq. from C

seq. from B

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

long branch attraction artifact



the two longest branches join together

seq. from A

seq. from D

seq. from C

seq. from B

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

What could you do to investigate if this is a possible explanation?

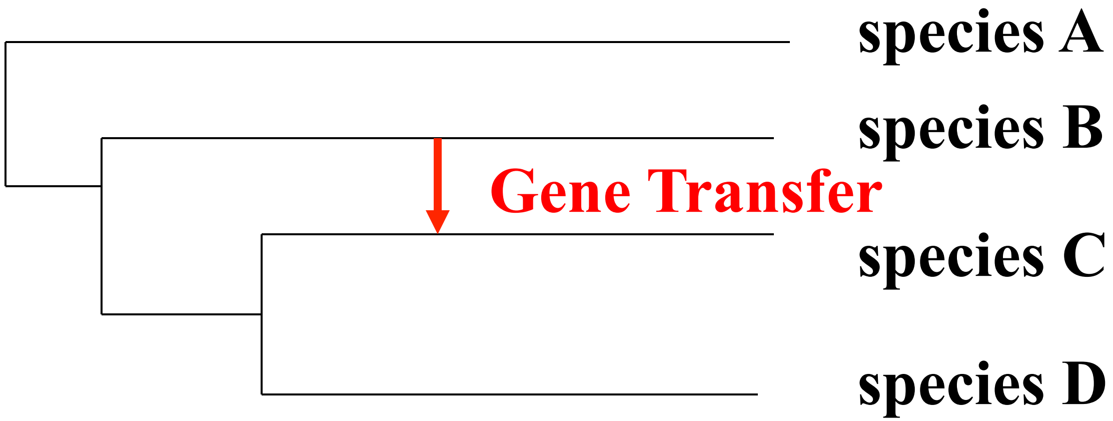
use only slow positions,

use an algorithm that corrects for ASRV

add sequences that break-up the long branches

Gene transfer

Organismal tree:



species A

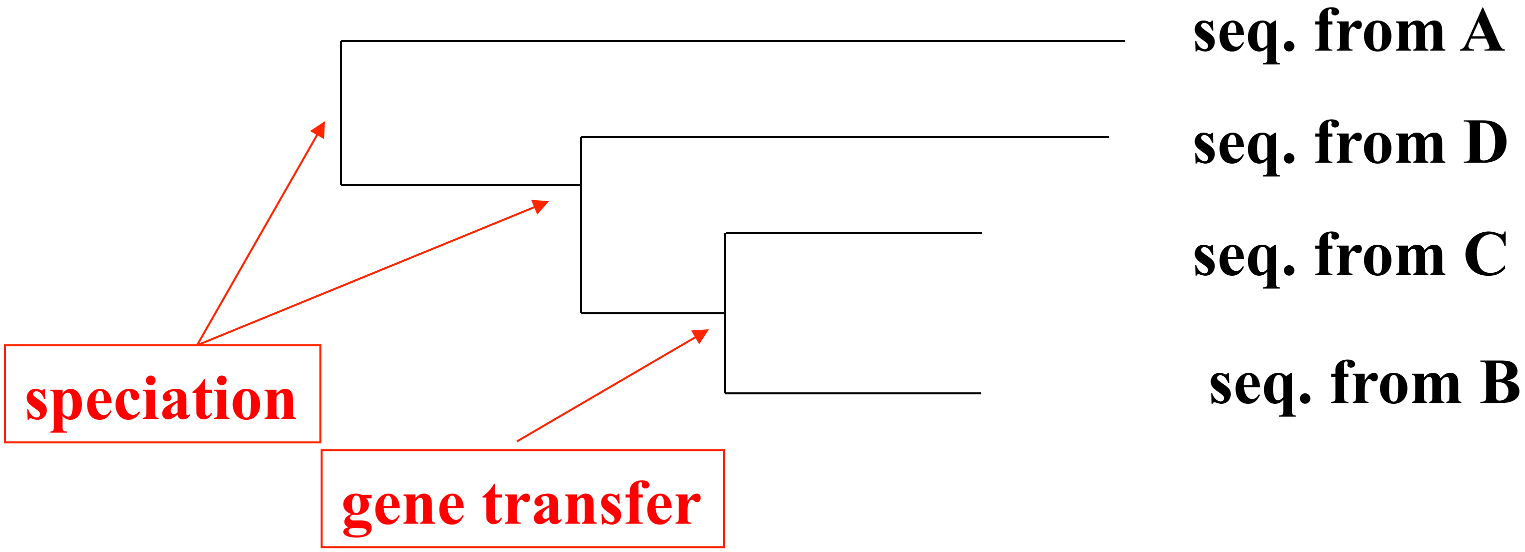
species B

Gene Transfer

species C

species D

molecular tree:



seq. from A

seq. from D

seq. from C

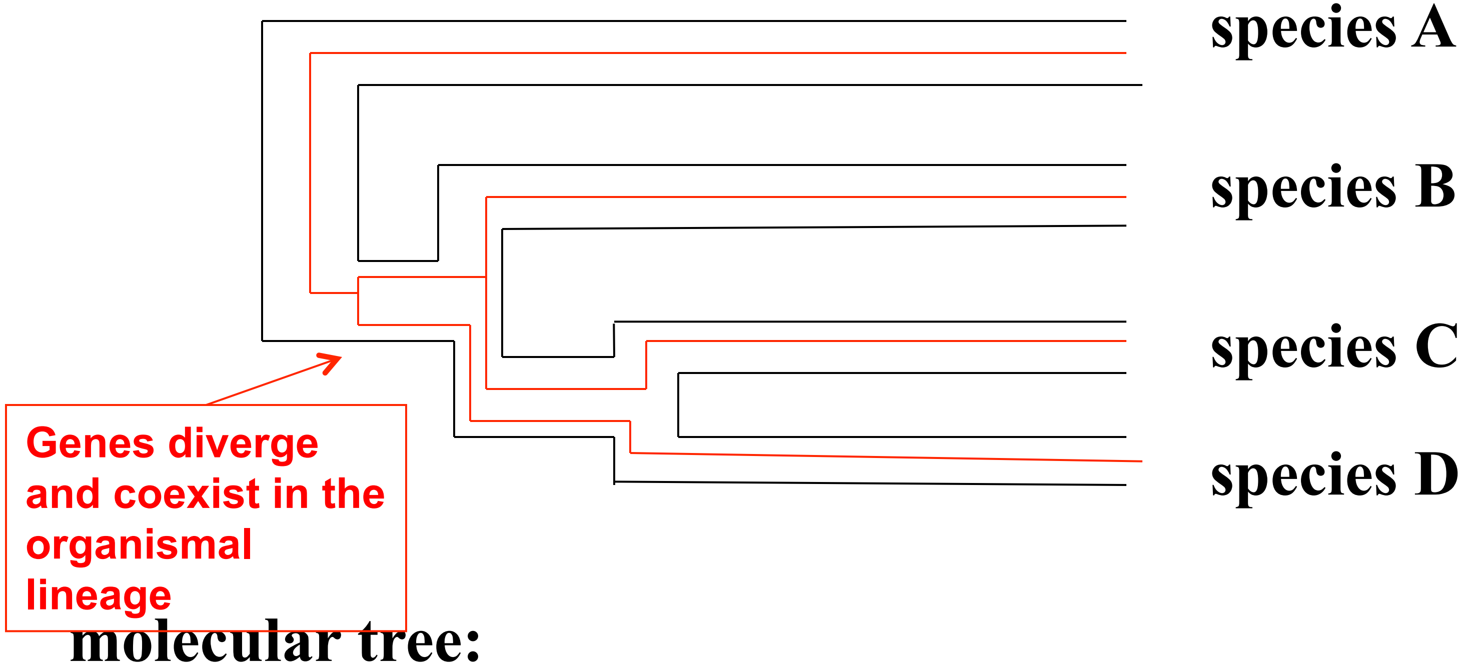
seq. from B

speciation

gene transfer

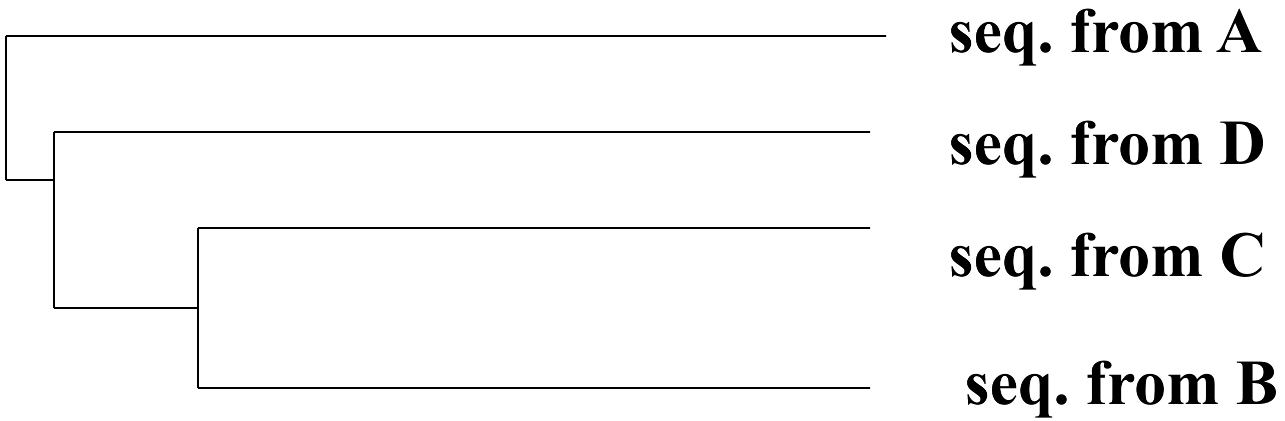
Lineage Sorting

Organismal tree:



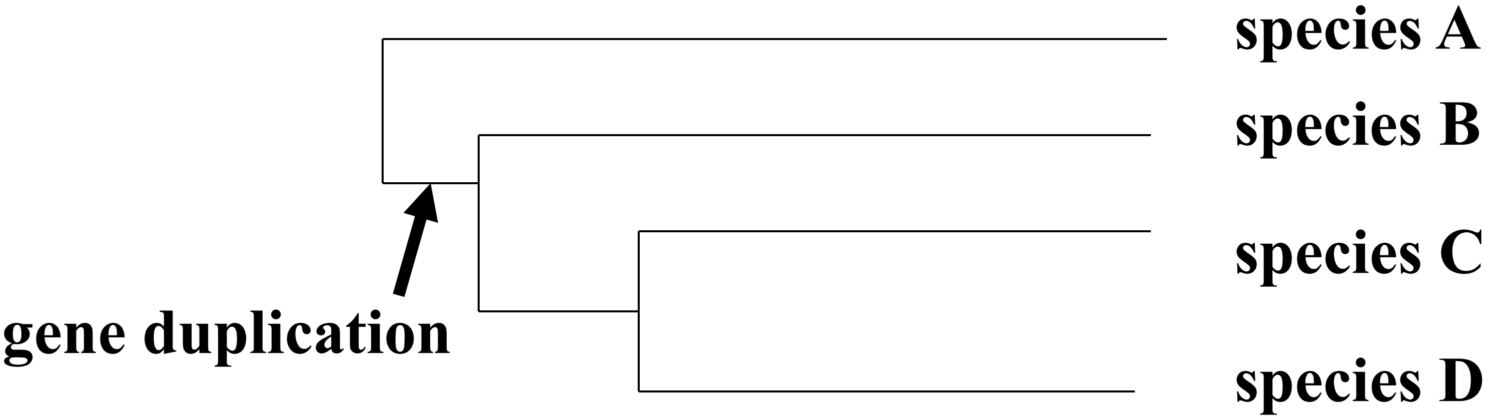
Genes diverge and coexist in the organismal lineage

molecular tree:

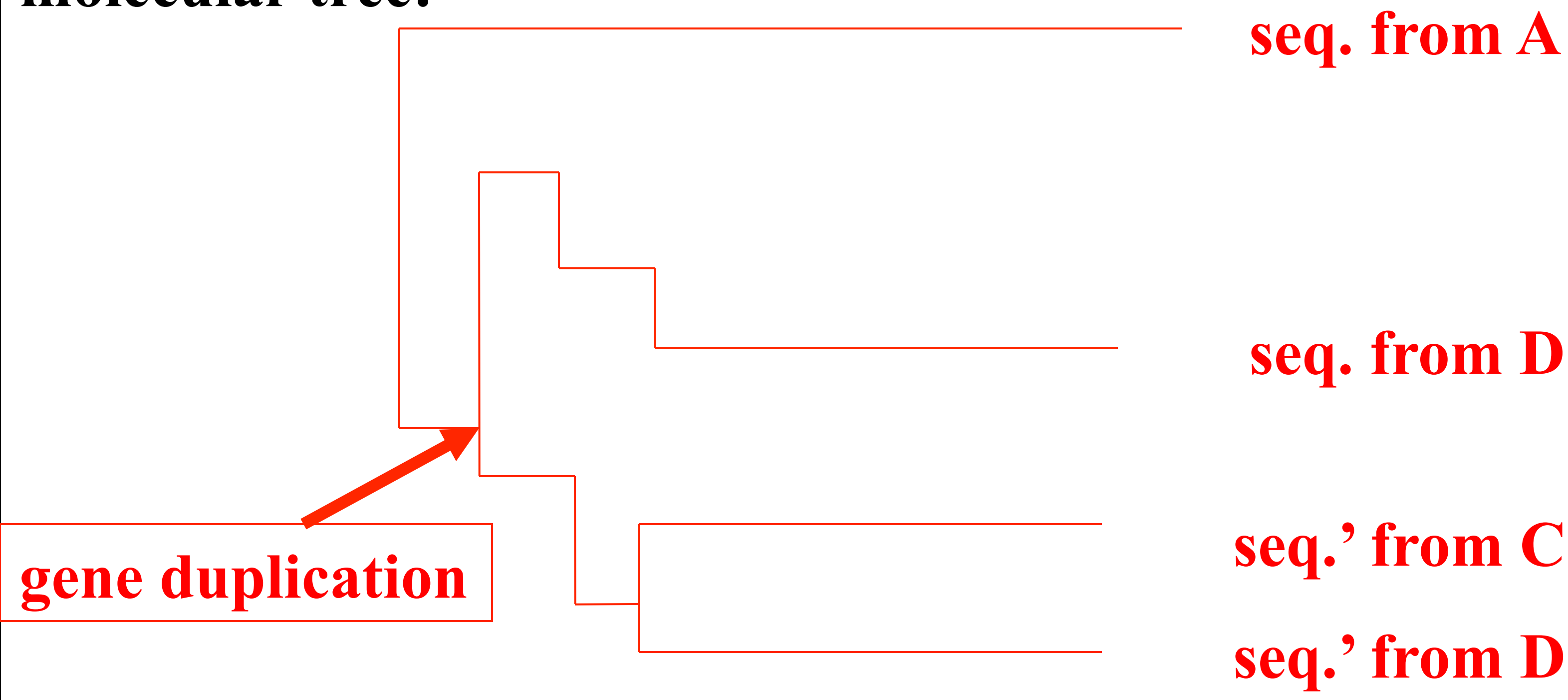


Gene duplication

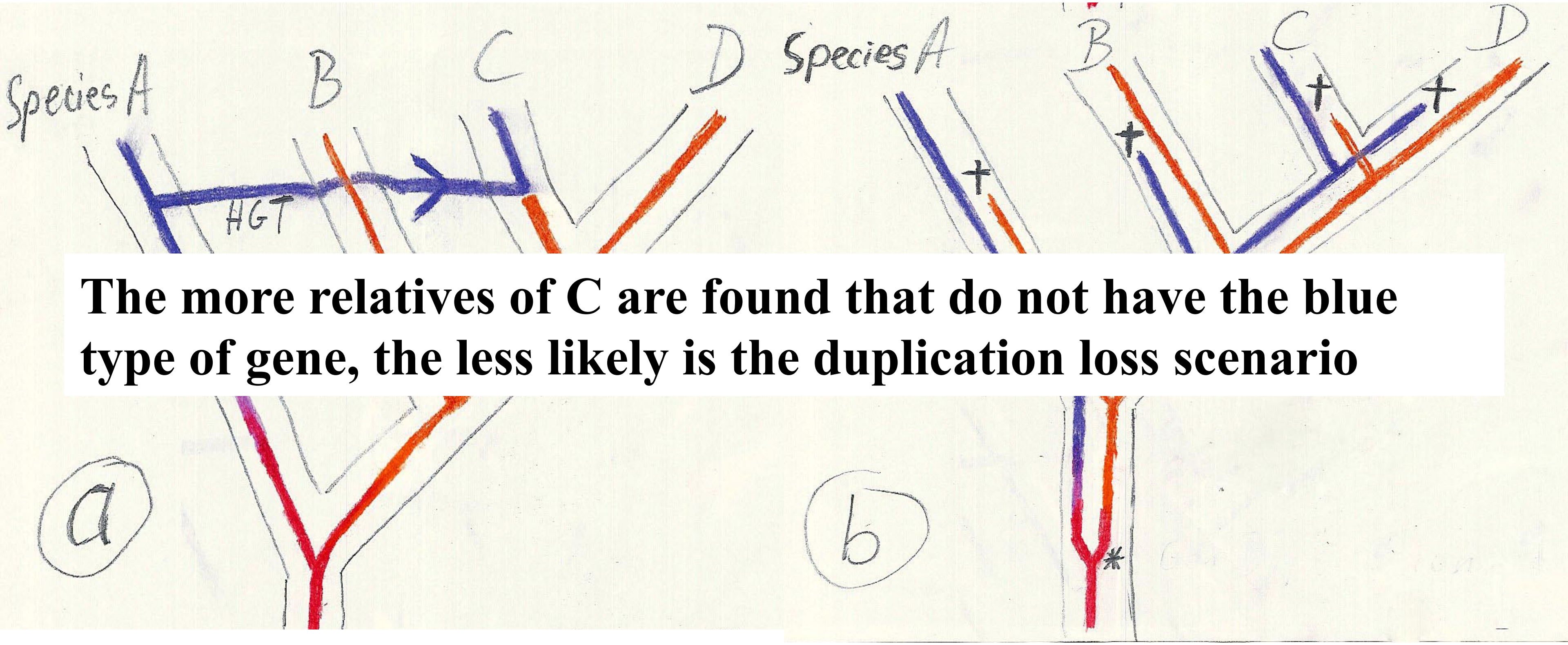
Organismal tree:



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

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

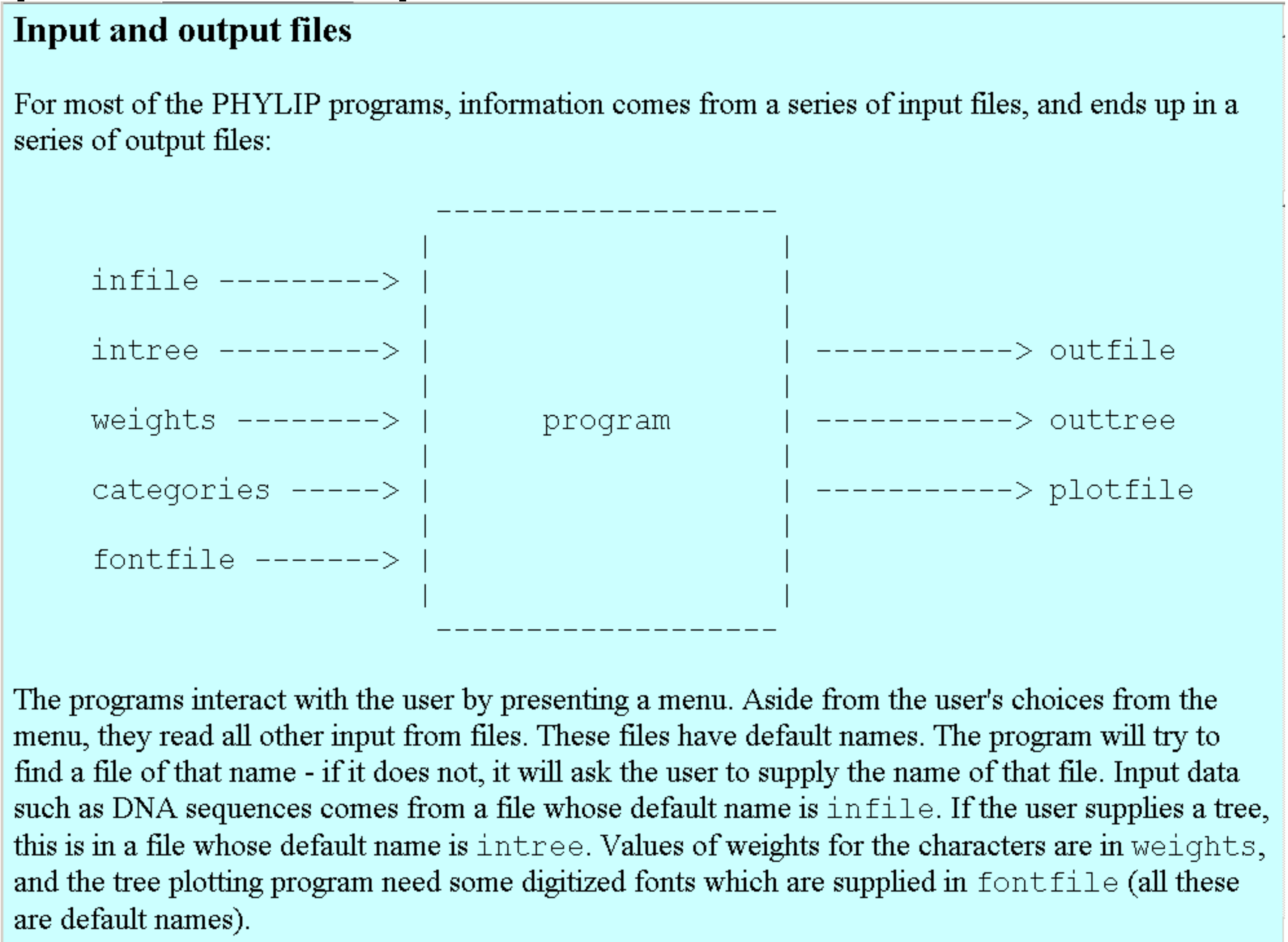
PHYLIP (the *PHYL*ogeny *I*nference *P*ackage) is a package of programs for inferring phylogenies (evolutionary trees).

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:

```
                      --------------------
                     |                    |
    infile --------->|                    |
                     |                    |
    intree --------->|                    |----------> outfile
                     |                    |
    weights -------->|      program       |----------> outtree
                     |                    |
    categories ----->|                    |----------> plotfile
                     |                    |
    fontfile ------->|                    |
                     |                    |
                      --------------------
```

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

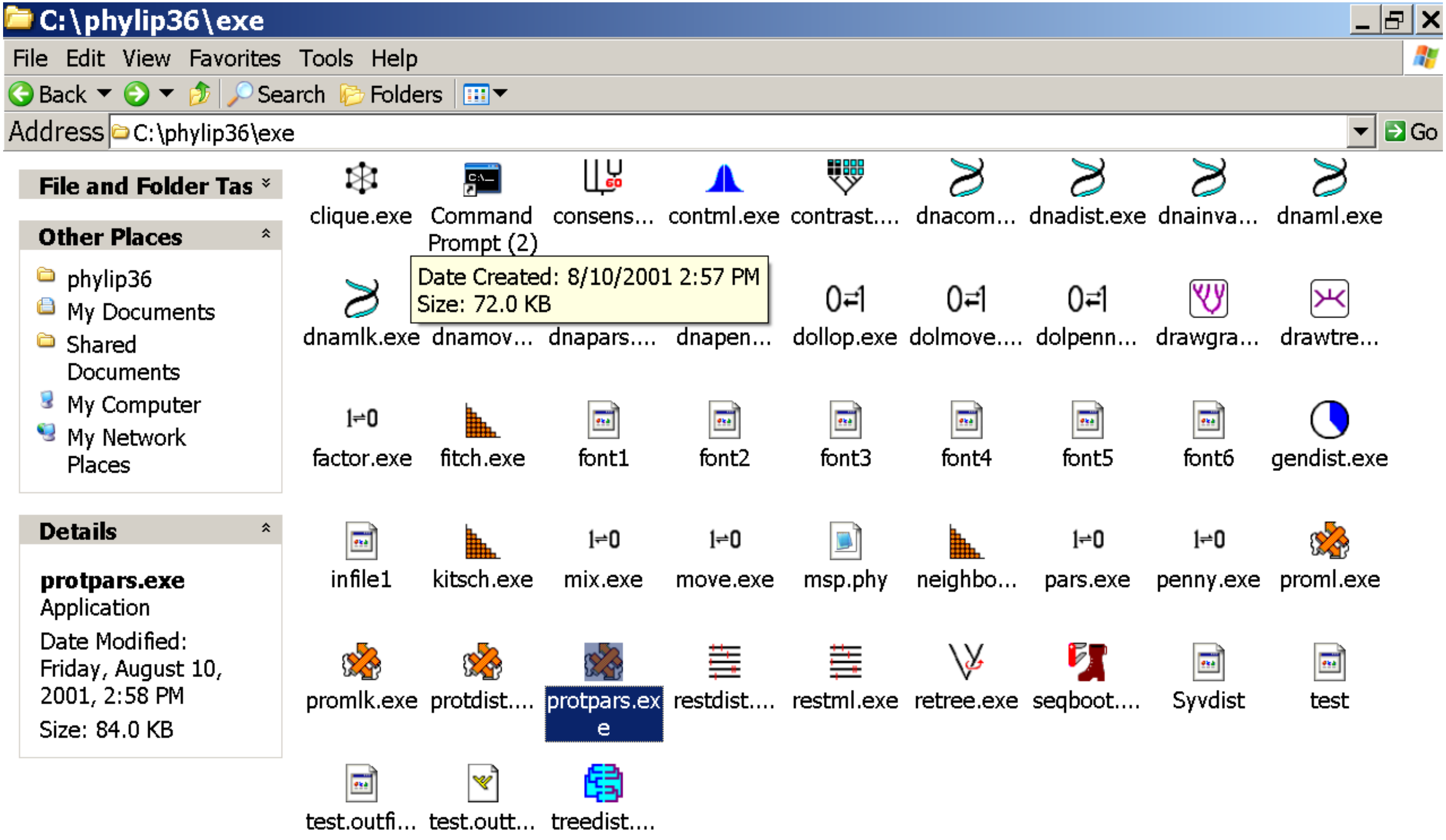
The easiest way to run PHYLIP programs is via a command line menu (similar to clustalw). The program is invoked through clicking on an icon, or by typing the program name at the command line.

```
> 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



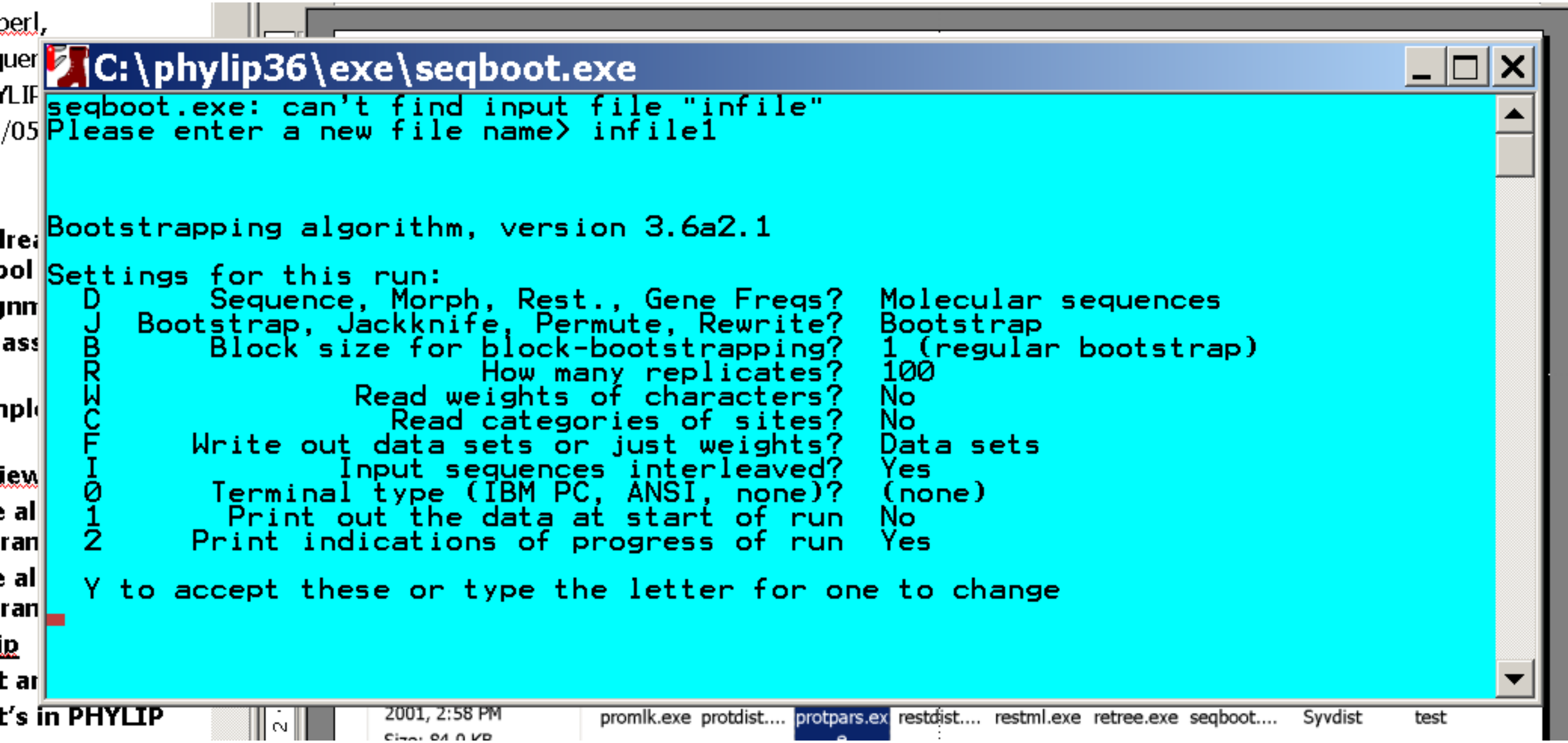
Date Created: 8/10/2001 2:57 PM
Size: 72.0 KB

protpars.exe
Application

Date Modified: Friday, August 10, 2001, 2:58 PM

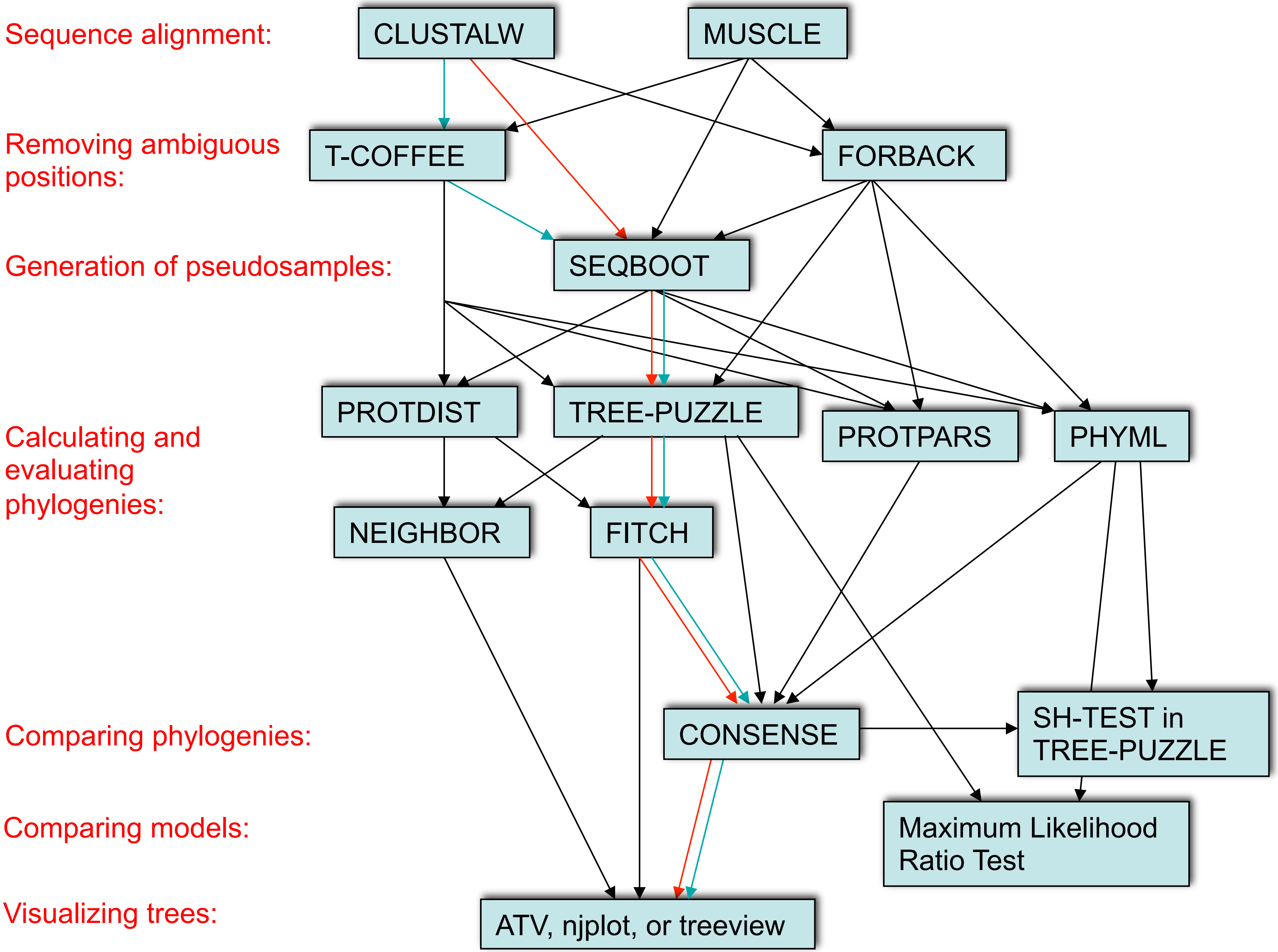
Size: 84.0 KB

menu interface



```
seqboot.exe: can't find input file "infile"
Please enter a new file name> infile1


Bootstrapping algorithm, version 3.6a2.1

Settings for this run:
  D      Sequence, Morph, Rest., Gene Freqs?  Molecular sequences
  J  Bootstrap, Jackknife, Permute, Rewrite?  Bootstrap
  B      Block size for block-bootstrapping?  1 (regular bootstrap)
  R                     How many replicates?  100
  W              Read weights of characters?  No
  C                Read categories of sites?  No
  F     Write out data sets or just weights?  Data sets
  I             Input sequences interleaved?  Yes
  0      Terminal type (IBM PC, ANSI, none)?  (none)
  1       Print out the data at start of run  No
  2     Print indications of progress of run  Yes

  Y to accept these or type the letter for one to change
```

example: seqboot and protpars on infile1



Sequence alignment: CLUSTALW MUSCLE

Removing ambiguous positions: T-COFFEE FORBACK

Generation of pseudosamples: SEQBOOT

Calculating and evaluating phylogenies: PROTDIST TREE-PUZZLE PROTPARS PHYML NEIGHBOR FITCH

Comparing phylogenies: CONSENSE SH-TEST in TREE-PUZZLE

Comparing models: Maximum Likelihood Ratio Test

Visualizing trees: ATV, njplot, or treeview

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

Example 1 Protpars

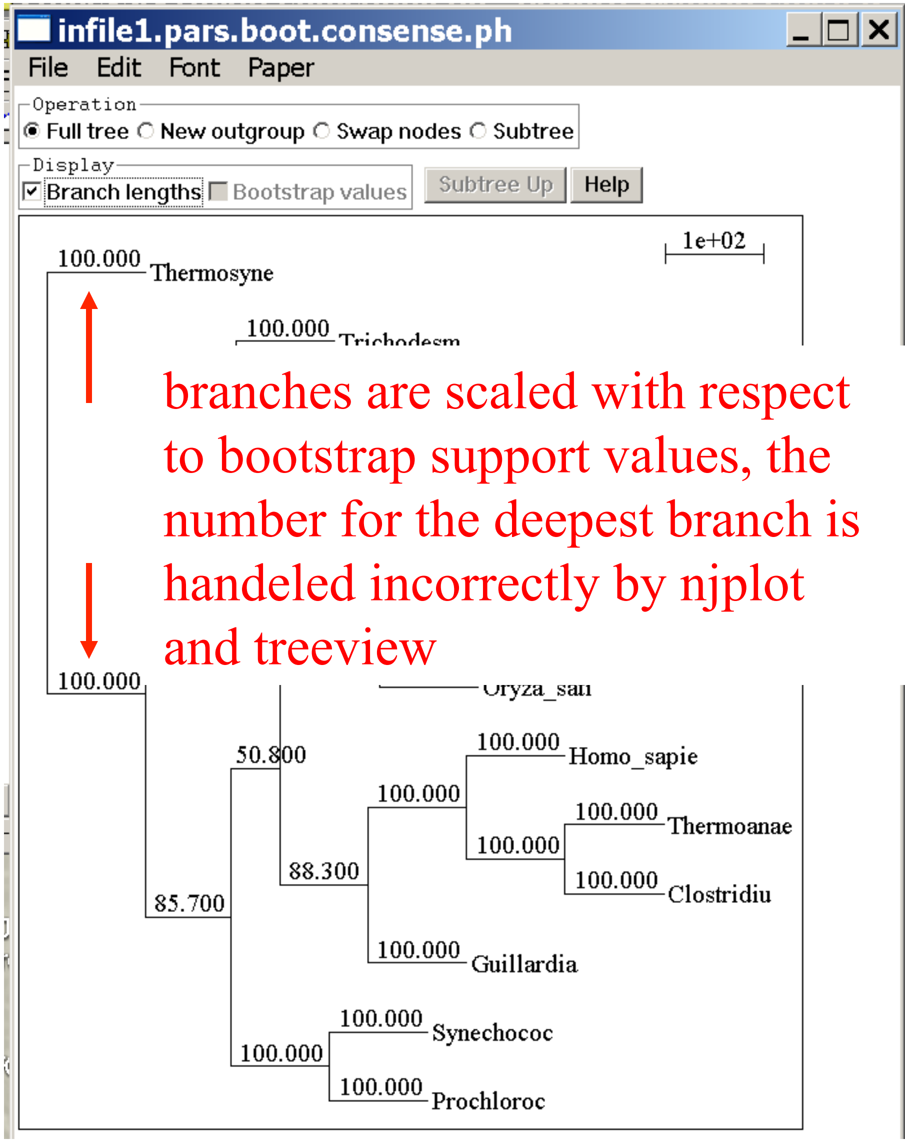
example: seqboot, protpars, consense on infile1

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)

```
Extended majority rule consensus tree

CONSENSUS TREE:
the numbers on the branches indicate the number
of times the partition of the species into the two sets
which are separated by that branch occurred
among the trees, out of 100.00 trees

                                            +------Prochloroc
                +----------------------100.-|
                |                           +------Synechococ
                |
                |             +--------------------Guillardia
         +-85.7-|             |
         |      |      +-88.3-|             +------Clostridiu
         |      |      |      |      +-100.-|
         |      |      |      +-100.-|      +------Thermoanae
         |      +-50.8-|             |
         |             |             +-------------Homo sapie
  +------|             |
  |      |             |                    +------Oryza sati
  |      |             +---------------100.0-|
  |      |                                  +------Arabidopsi
  |      |
  |      |                    +--------------------Synechocys
  |      |                    |
  |      +---------------53.0-|             +------Nostoc pun
  |                           |      +-99.5-|
  |                           +-38.5-|      +------Nostoc sp
  |                                  |
  |                                  +-------------Trichodesm
  |
  +------------------------------------------------Thermosyne


  remember: this is an unrooted tree!
```



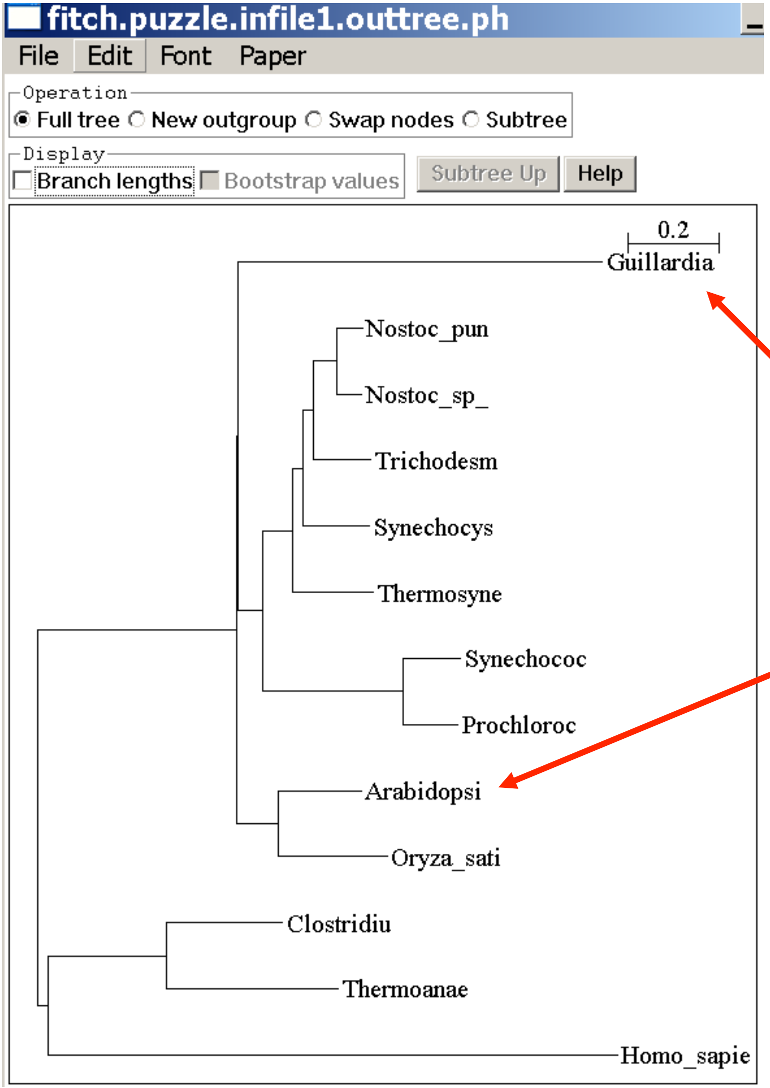
infile1.pars.boot.consense.ph

Operation: ● Full tree ○ New outgroup ○ Swap nodes ○ Subtree

Display: ☑ Branch lengths □ Bootstrap values

branches are scaled with respect to bootstrap support values, the number for the deepest branch is handeled incorrectly by njplot and treeview

(protpars versus) distance/FM



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

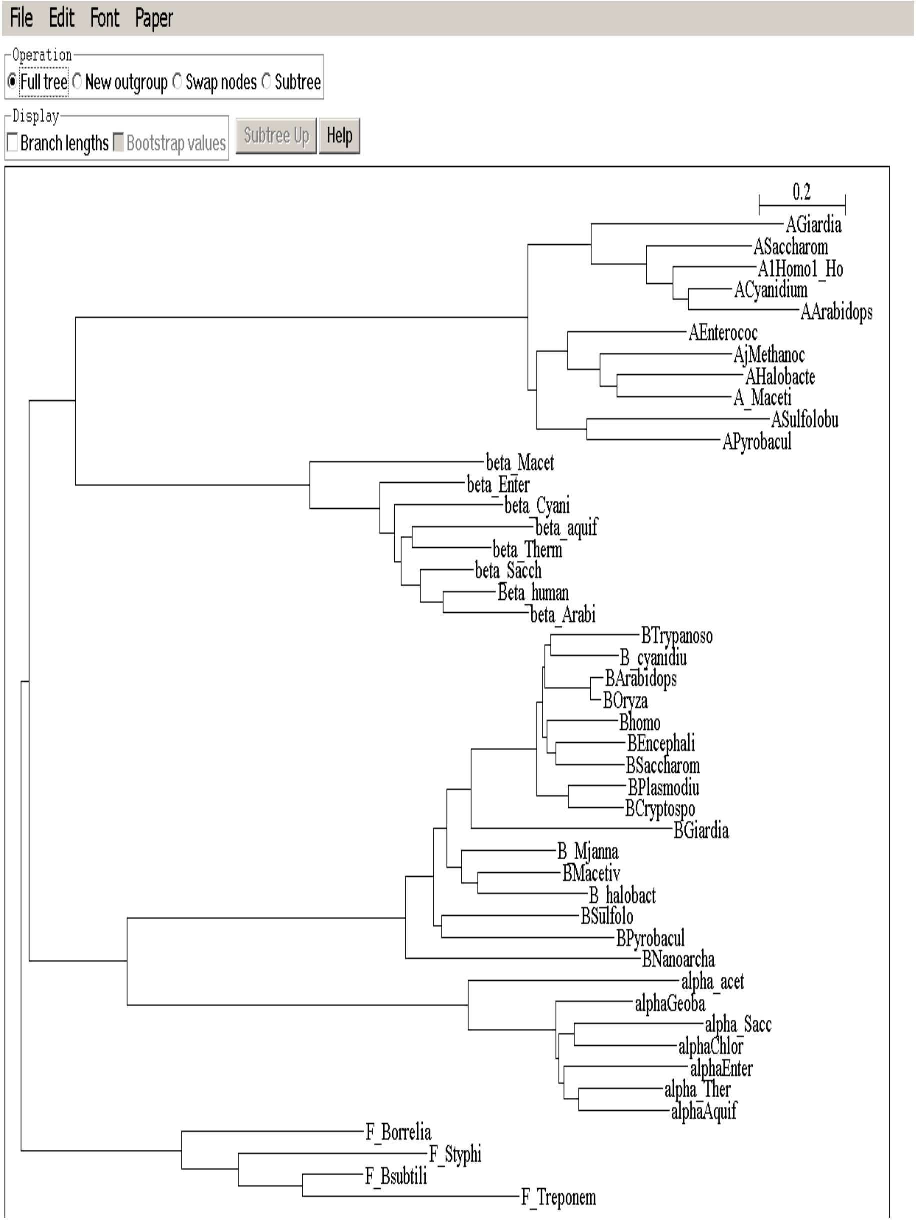
what might be the explanation for the red algae not grouping with the plants?

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



Operation
● Full tree ○ New outgroup ○ Swap nodes ○ Subtree

Display
□ Branch lengths □ Bootstrap values

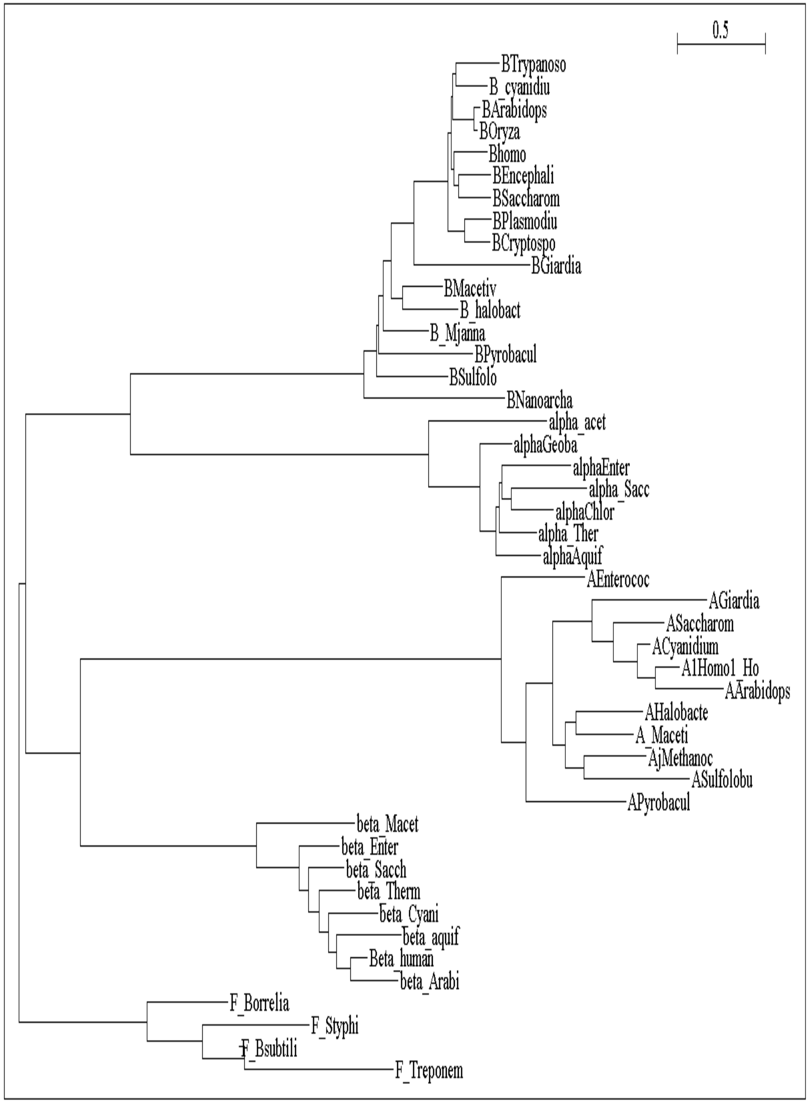
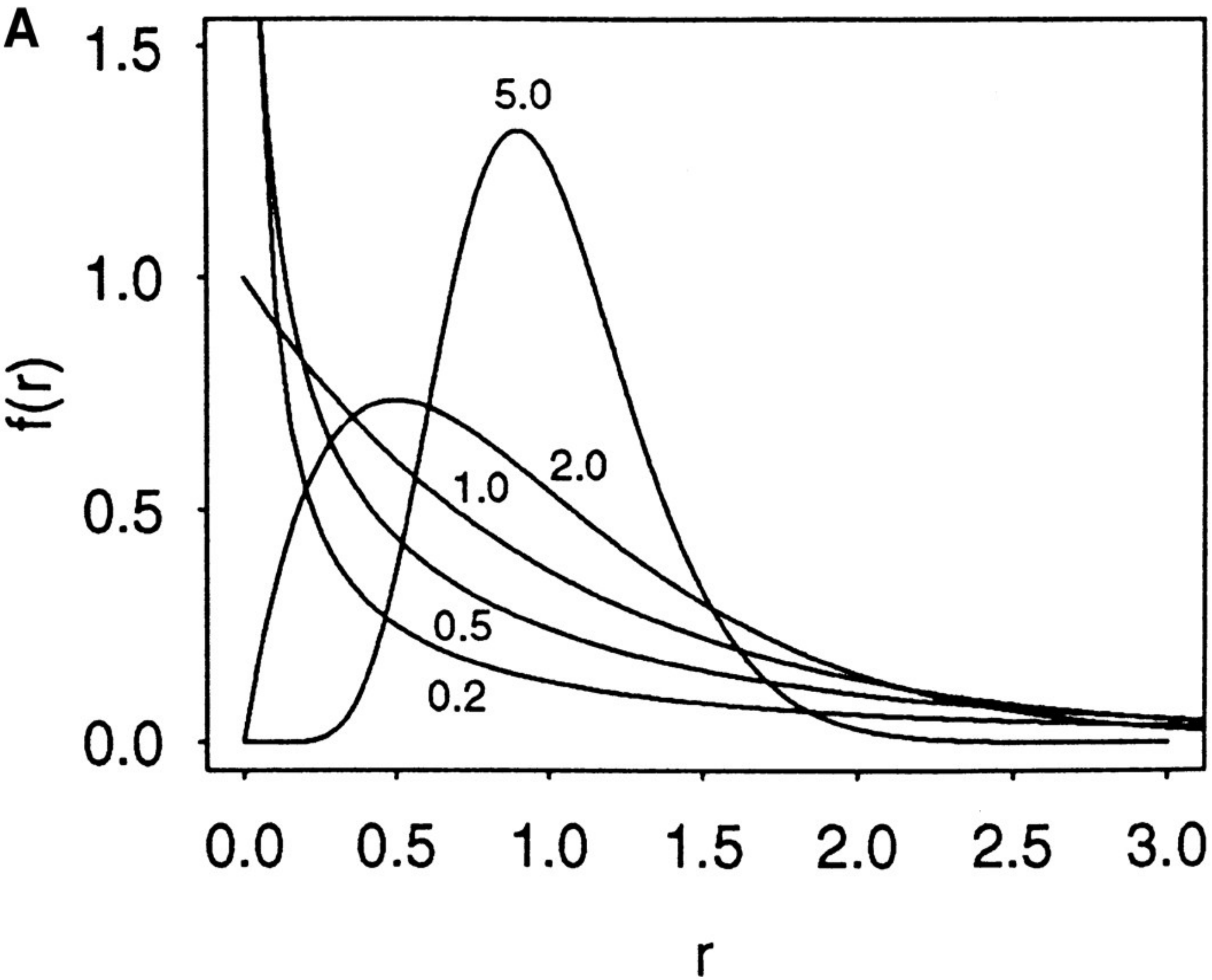
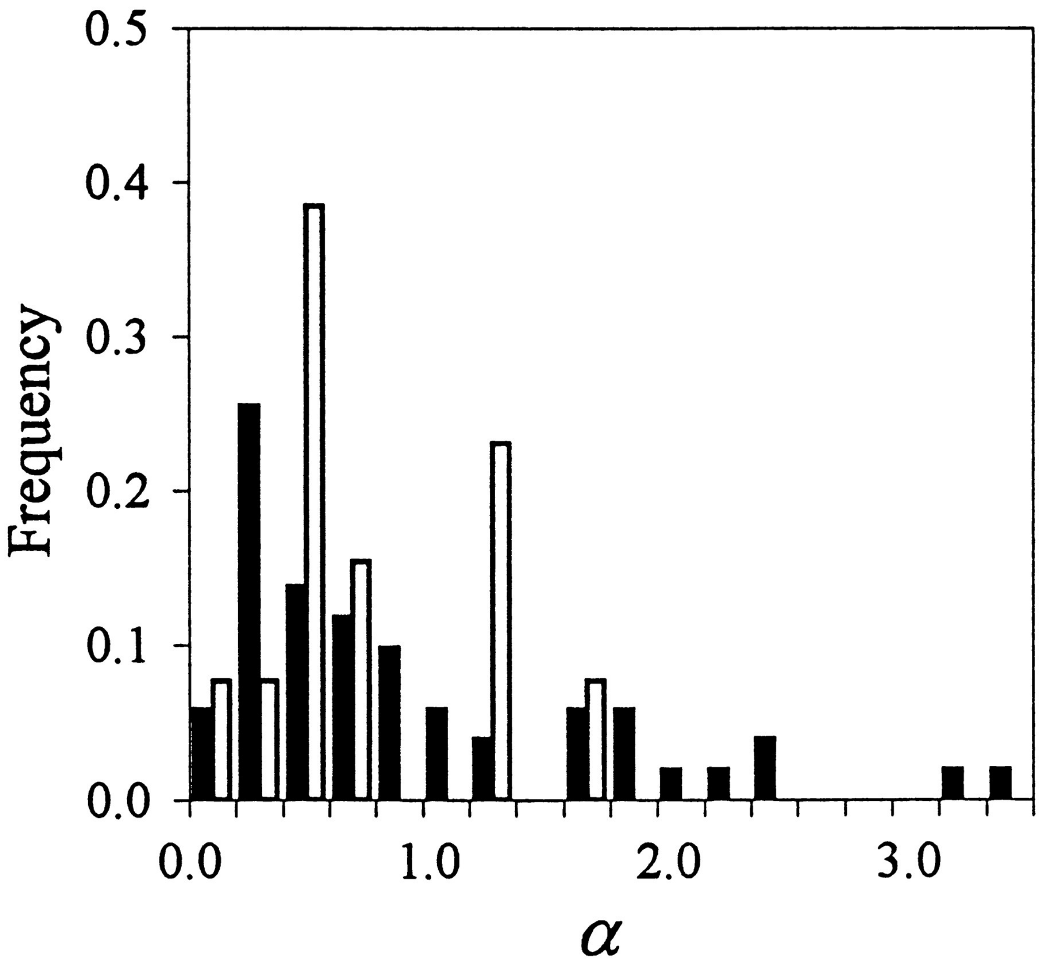


Figure 1. The probability density functions of gamma distributions



Zhang, J. et al. Genetics 1998;149:1615-1625

Figure 4. Distributions of the {alpha} values of 51 nuclear (solid histograms) and 13 mitochondrial (open histograms) genes

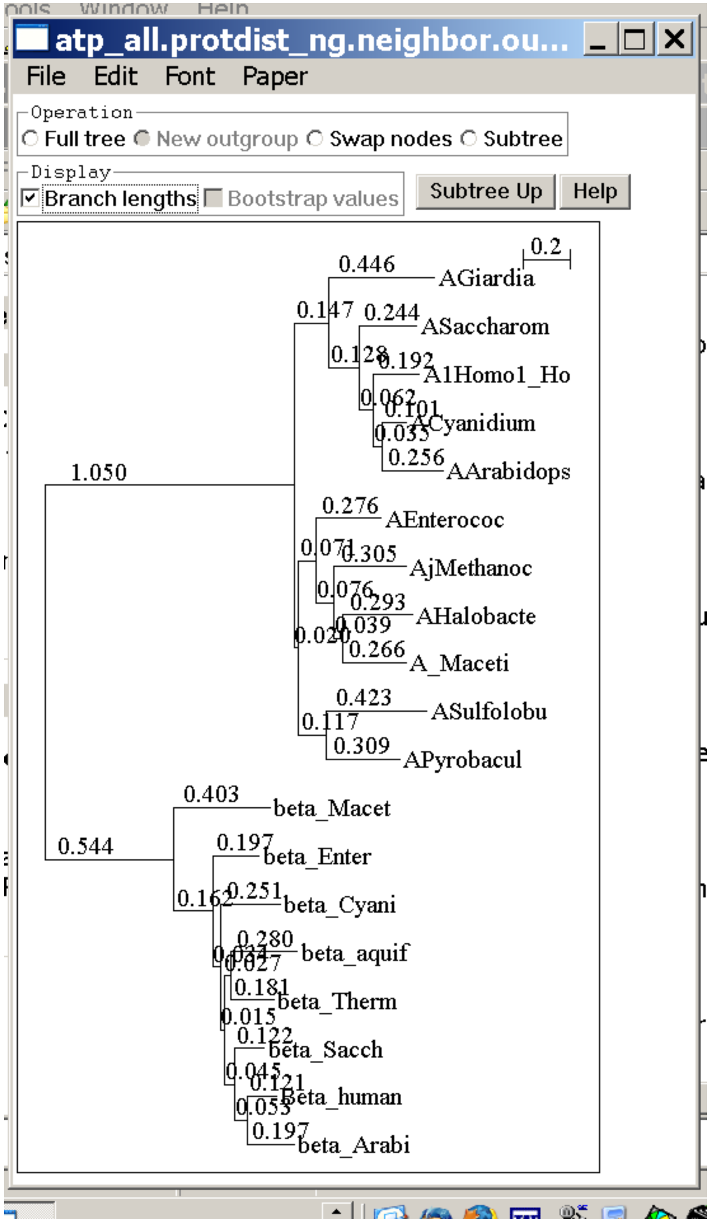


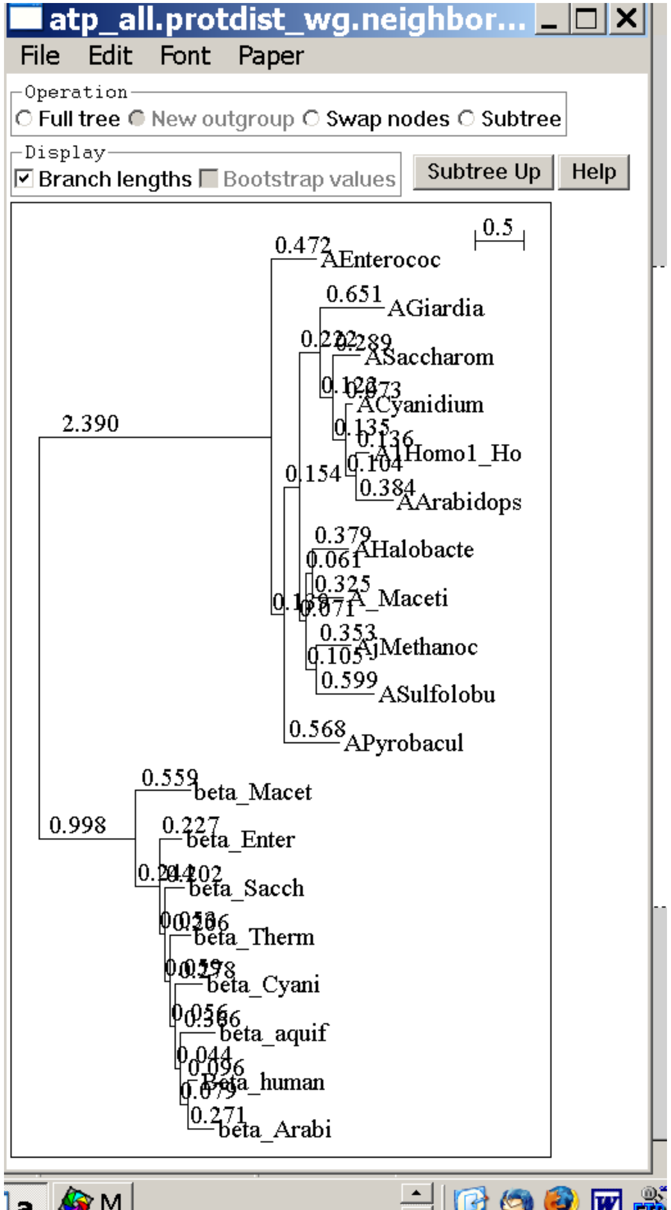
Zhang, J. et al. Genetics 1998;149:1615-1625

GENETICS

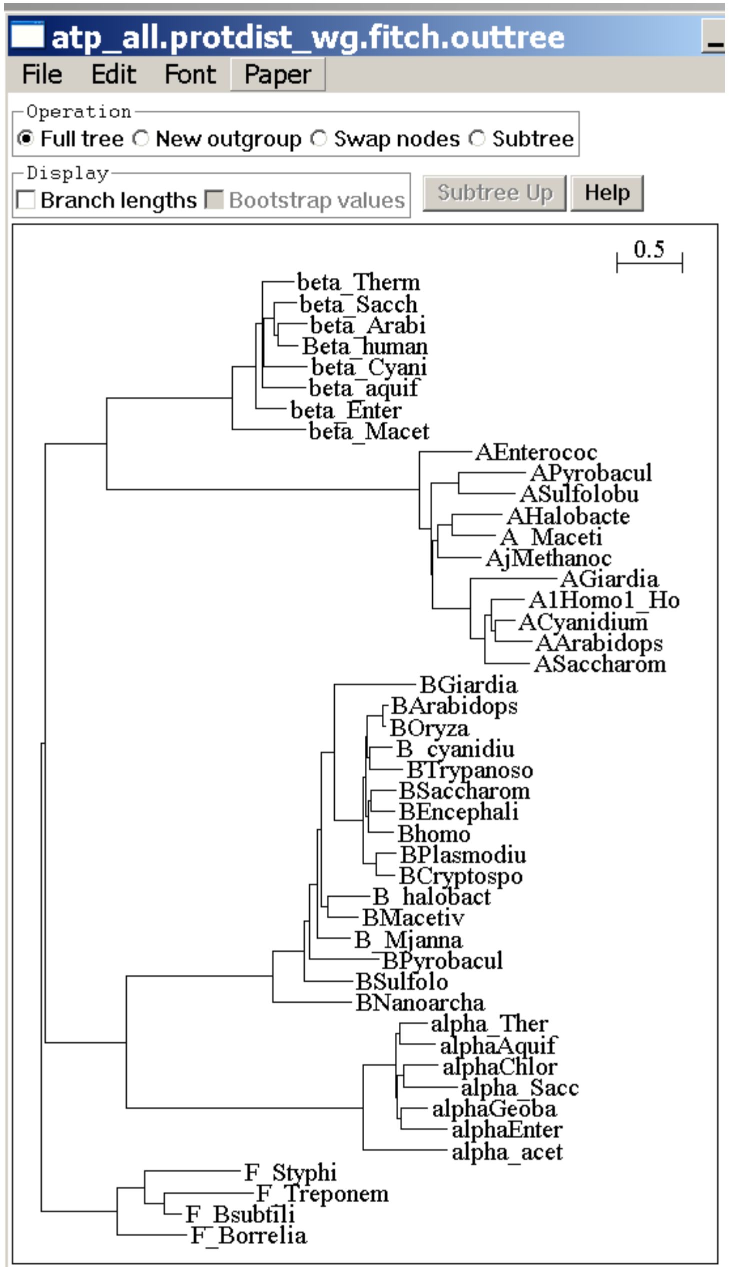
subtree with branch lengths

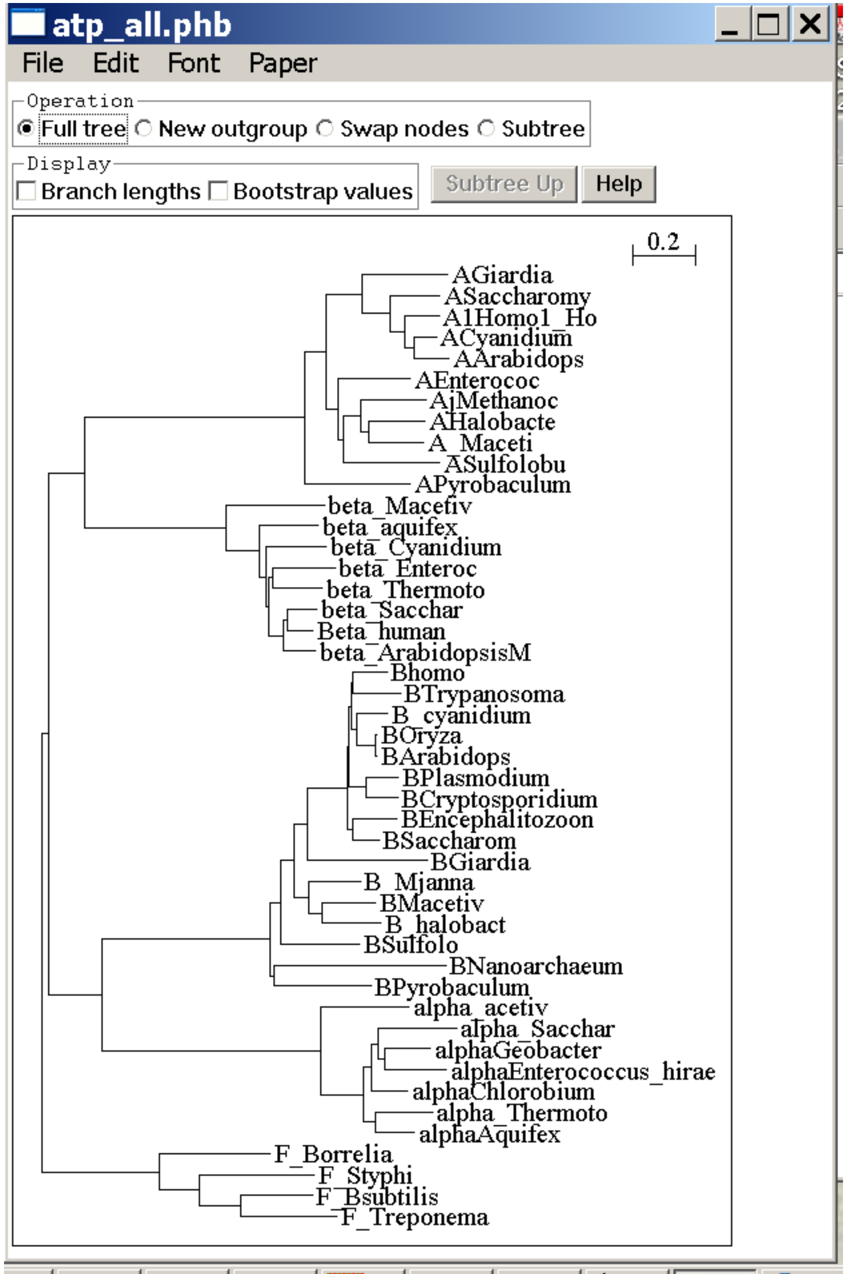
without and with correction for ASRV



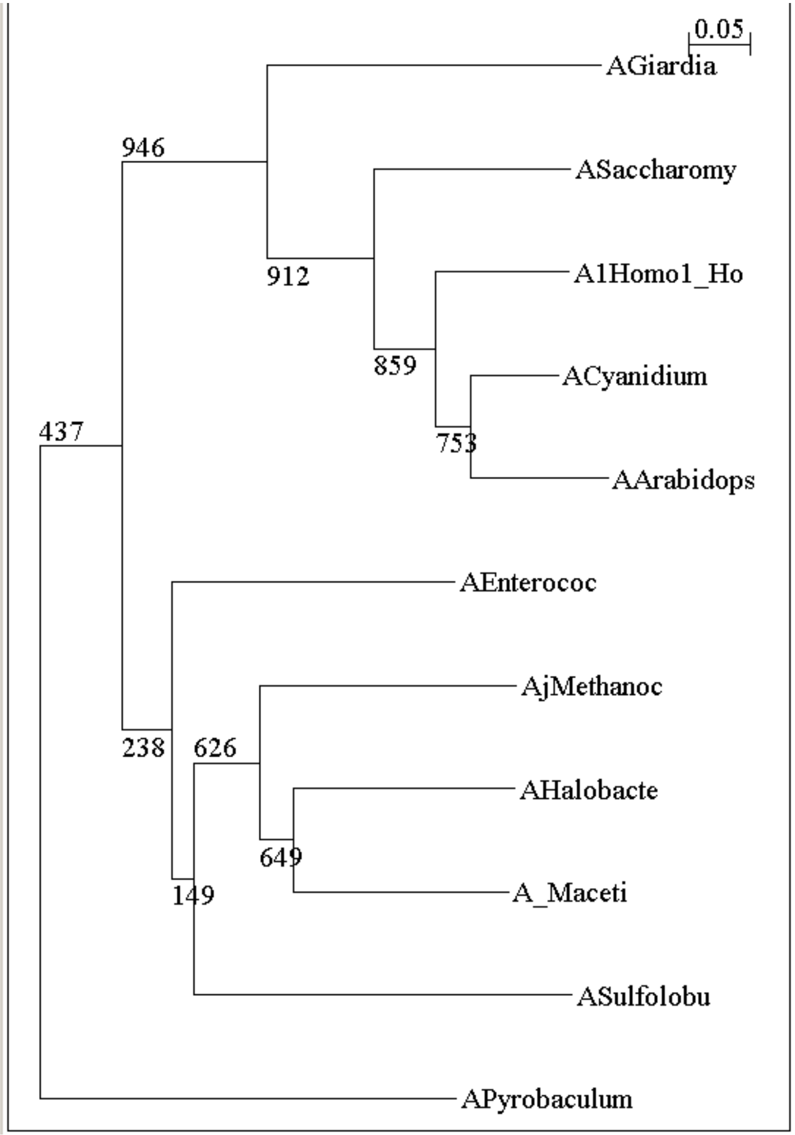


compare to trees with FITCH and clustalw – same dataset

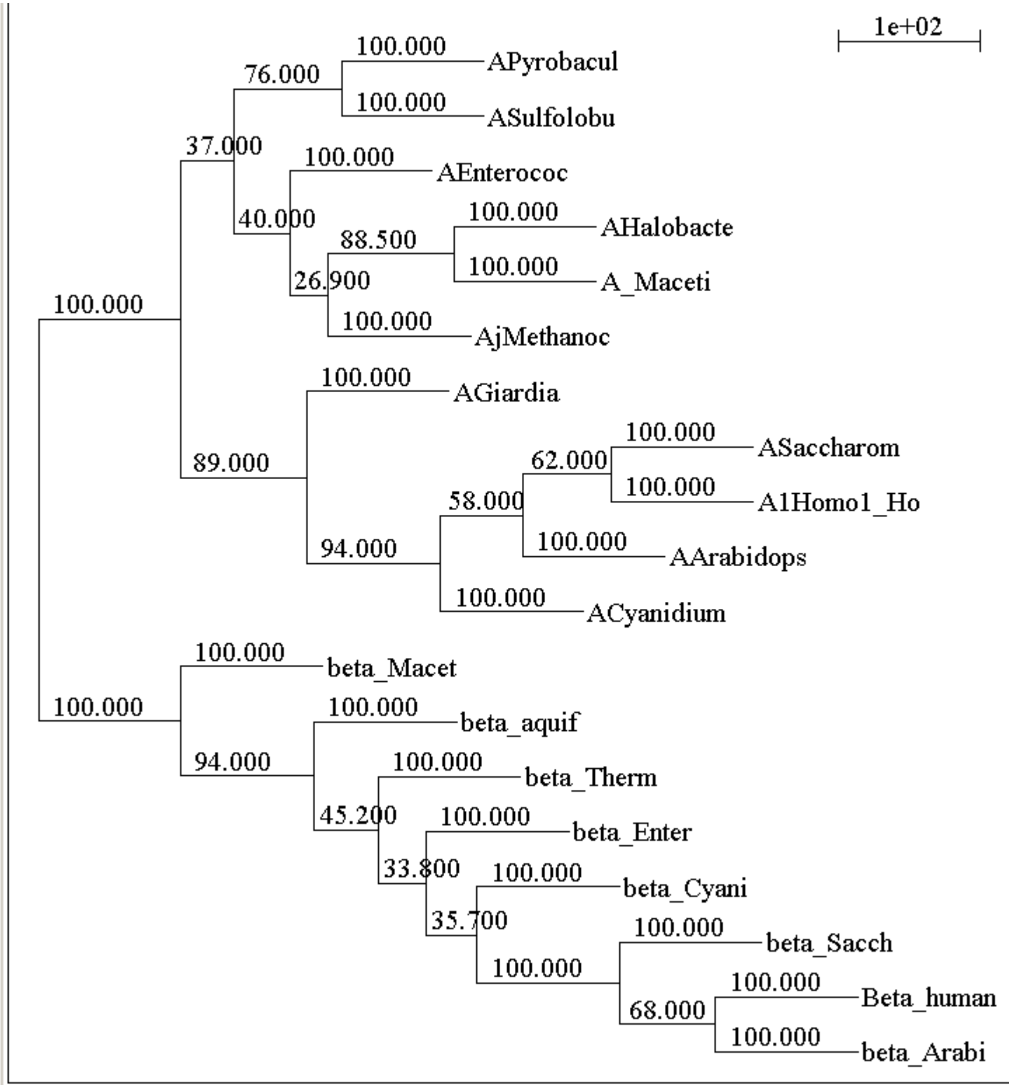




bootstrap support ala clustal



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.

Phyml is part of seaview4

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)

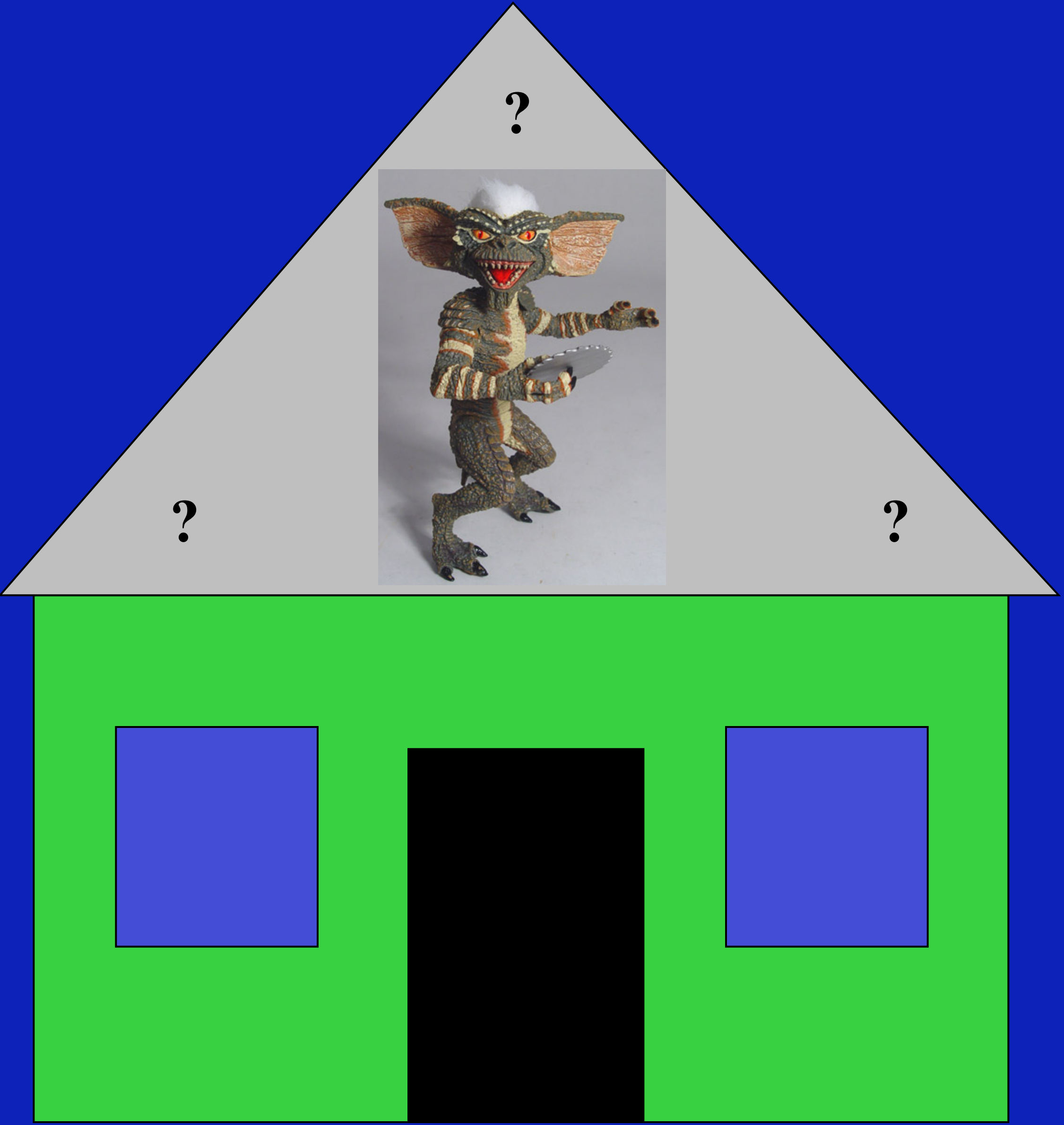
- TREEPUZZLE calculates distance matrices using the ml specified model. These can be used in FITCH or Neighbor. PUZZLEBOOT automates this approach to do bootstrap analyses – WARNING: this is a distance matrix analyses! The official script for PUZZLEBOOT is here – you need to create a command file (puzzle.cmds), 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! @@@

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.

Elliot Sober’s Gremlins



Observation: Loud noise in the attic

Hypothesis: *gremlins in the attic playing bowling*

Likelihood =
P(noise|gremlins in the attic)

P(gremlins in the attic|noise)

Bayes’ Theorem

Reverend Thomas Bayes
(1702-1761)

$$\underbrace{P(model|data, I)}_{\text{Posterior Probability}} = \underbrace{P(model, I)}_{\text{Prior Probability}} \frac{\overbrace{P(data|model, I)}^{\text{Likelihood}}}{\underbrace{P(data,I)}_{\text{Normalizing constant}}}$$

Likelihood

describes how well the model predicts the data

Posterior Probability

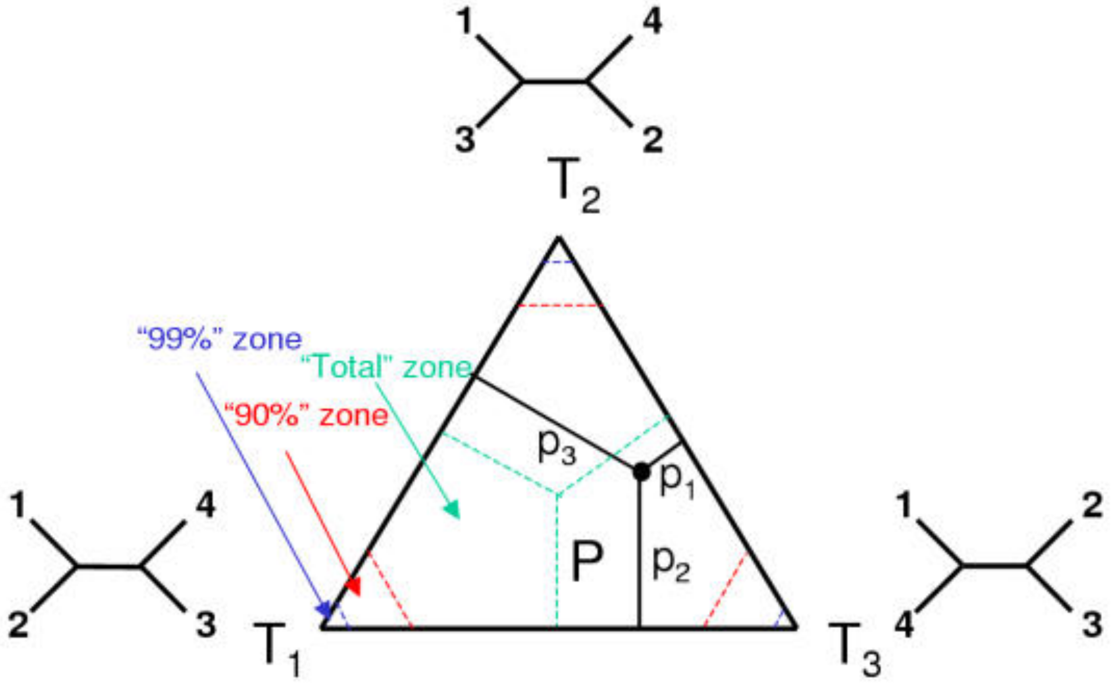
represents the degree to which we believe a given **model** accurately describes the situation given the available **data** and all of our prior information **I**

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

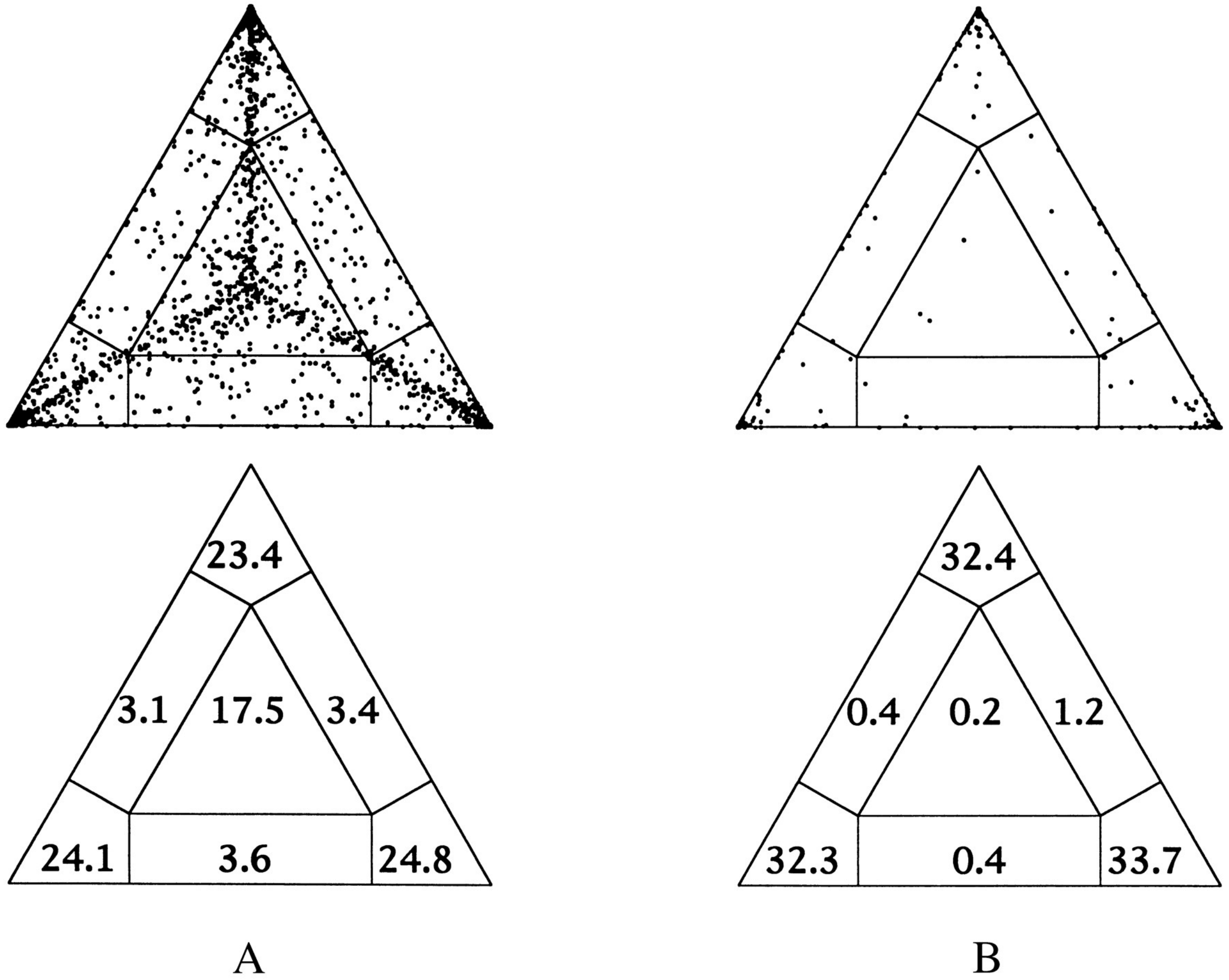
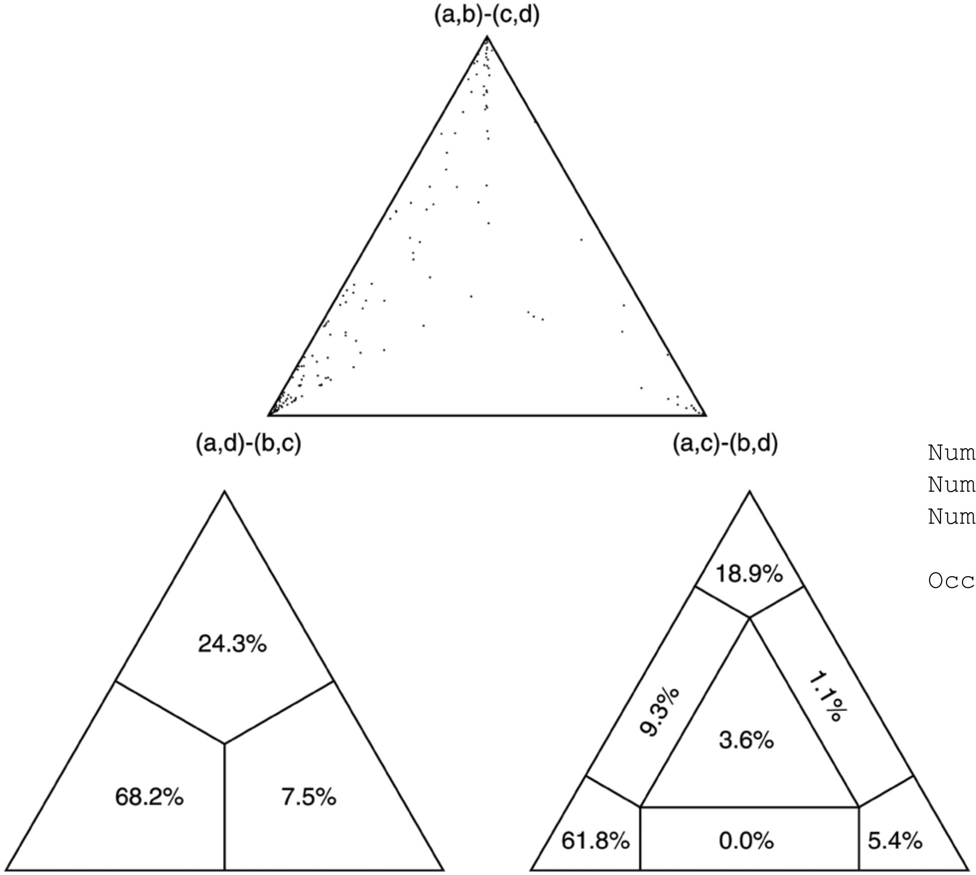


Figure 5. Likelihood-mapping analysis for two biological data sets. (*Upper*) The distribution patterns. (*Lower*) The occupancies (in percent) for the seven areas of attraction.

(*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



Cluster a: 14 sequences
outgroup (prokaryotes)

Cluster b: 20 sequences
other Eukaryotes

Cluster c: 1 sequences
Plasmodium

Cluster d: 1 sequences
Giardia

```
               (a,b)-(c,d)
                   /\
                  /  \
                 /    \
                /   1  \
               / \    / \
              /   \  /   \
             /     \/     \
            /  3    :   2  \
           /        :       \
          /__________________\
(a,d)-(b,c)                  (a,c)-(b,d)

Number of quartets in region 1: 68 (= 24.3%)
Number of quartets in region 2: 21 (= 7.5%)
Number of quartets in region 3: 191 (= 68.2%)

Occupancies of the seven areas 1, 2, 3, 4, 5, 6, 7:

               (a,b)-(c,d)
                   /\
                  /  \
                 /  1 \
                / \  / \
               /   /\   \
              / 6 /  \ 4 \
             /   /  7 \   \
            / \ /______\ / \
           / 3  :   5  :  2 \
          /__________________\
(a,d)-(b,c)                  (a,c)-(b,d)

Number of quartets in region 1: 53 (= 18.9%)
Number of quartets in region 2: 15 (= 5.4%)
Number of quartets in region 3: 173 (= 61.8%)
Number of quartets in region 4: 3 (= 1.1%)
Number of quartets in region 5: 0 (= 0.0%)
Number of quartets in region 6: 26 (= 9.3%)
Number of quartets in region 7: 10 (= 3.6%)
```

Alternative Approaches to Estimate Posterior Probabilities

Bayesian Posterior Probability Mapping with MrBayes

(Huelsenbeck and Ronquist, 2001)

Problem:

Strimmer's formula $p_i = \dfrac{L_i}{L_1+L_2+L_3}$

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

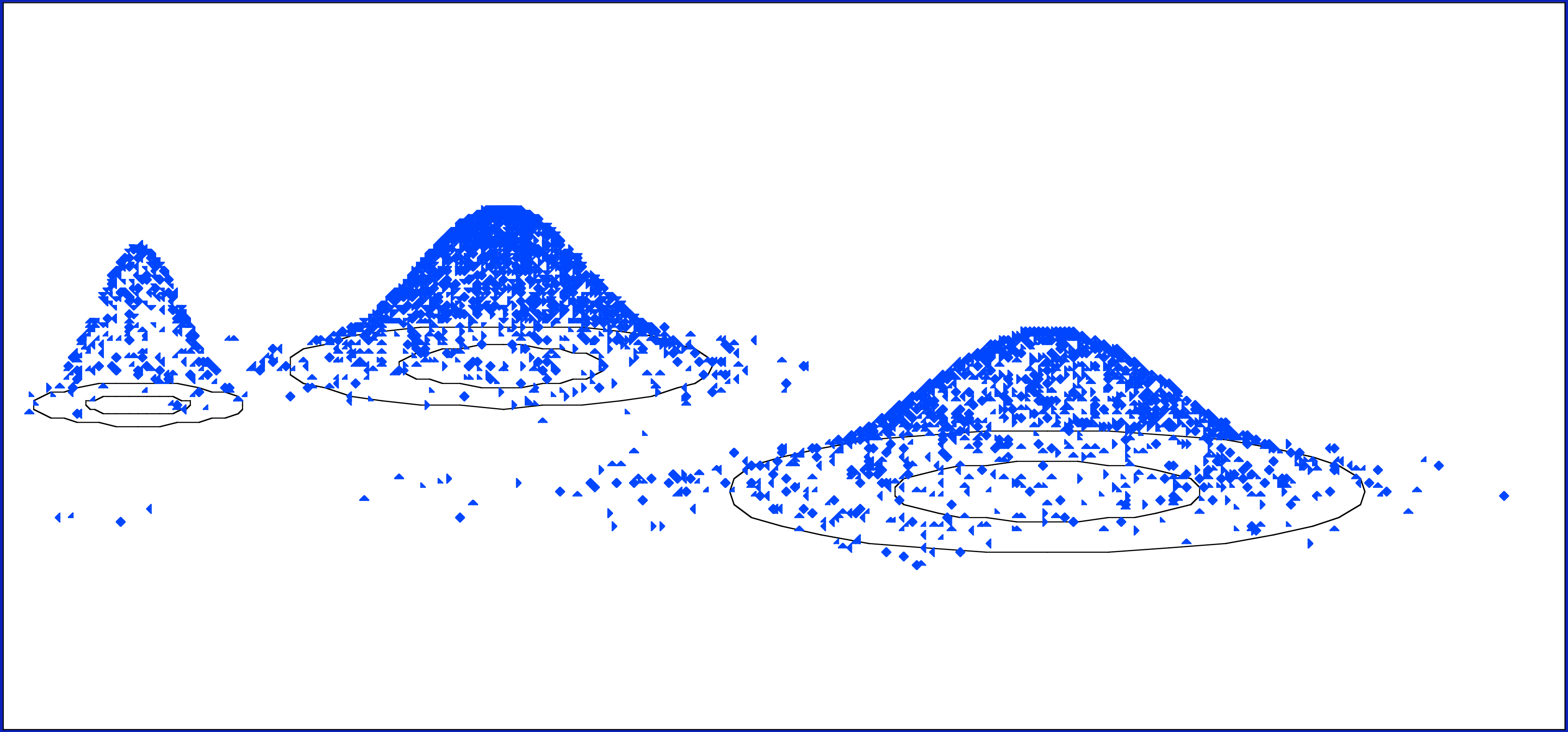
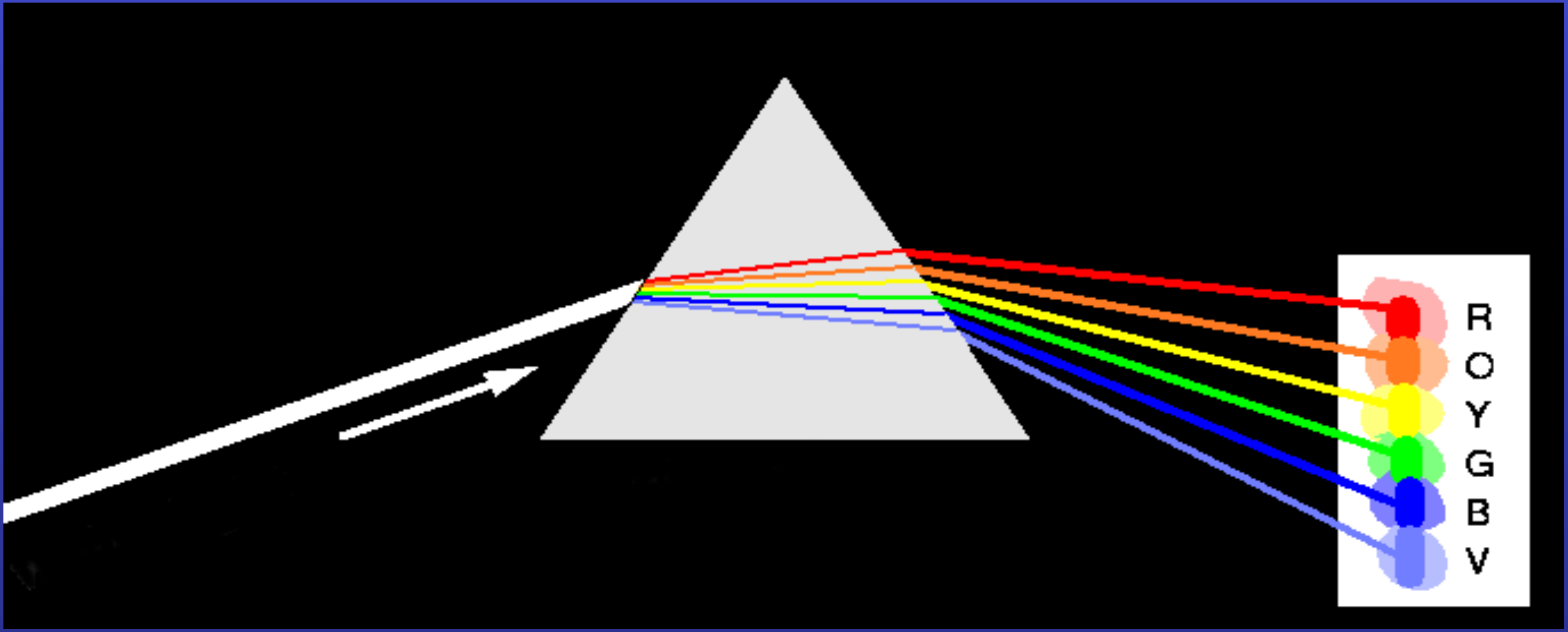


Figure generated using MCRobot program (Paul Lewis, 2001)

Decomposition of Phylogenetic Data



Phylogenetic information present in genomes

Break information into small quanta of information (bipartitions or embedded quartets)

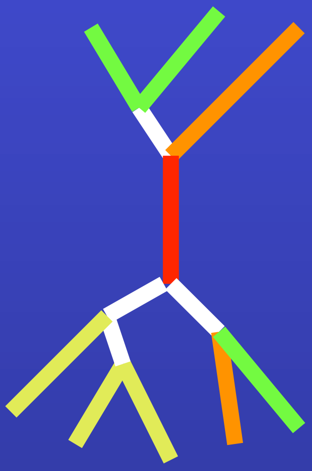
Analyze spectra to detect transferred genes and plurality consensus.

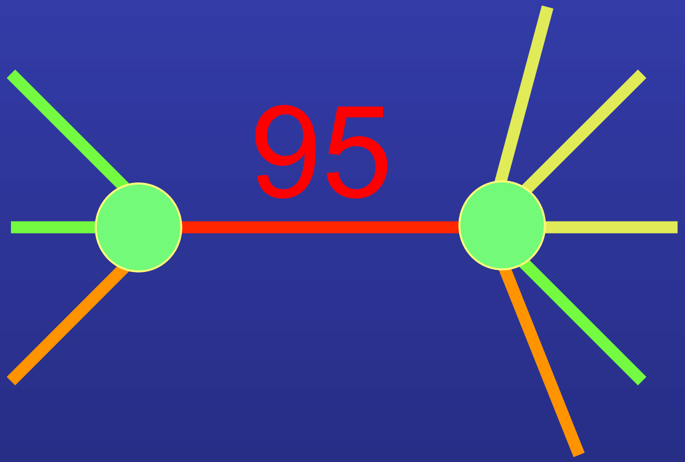
TOOLS TO ANALYZE PHYLOGENETIC INFORMATION FROM MULTIPLE GENES IN GENOMES:

Bipartition Spectra (Lento Plots)

BIPARTITION OF A PHYLOGENETIC TREE

Bipartition (or split) – a division of a phylogenetic tree into two parts that are connected by a single branch.
It divides a dataset into two groups, but it does not consider the relationships within each of the two groups.





* * * . . . . .

Yellow vs Rest

* * * . . . * *

compatible to illustrated bipartition

Orange vs Rest

. . * . . . . *

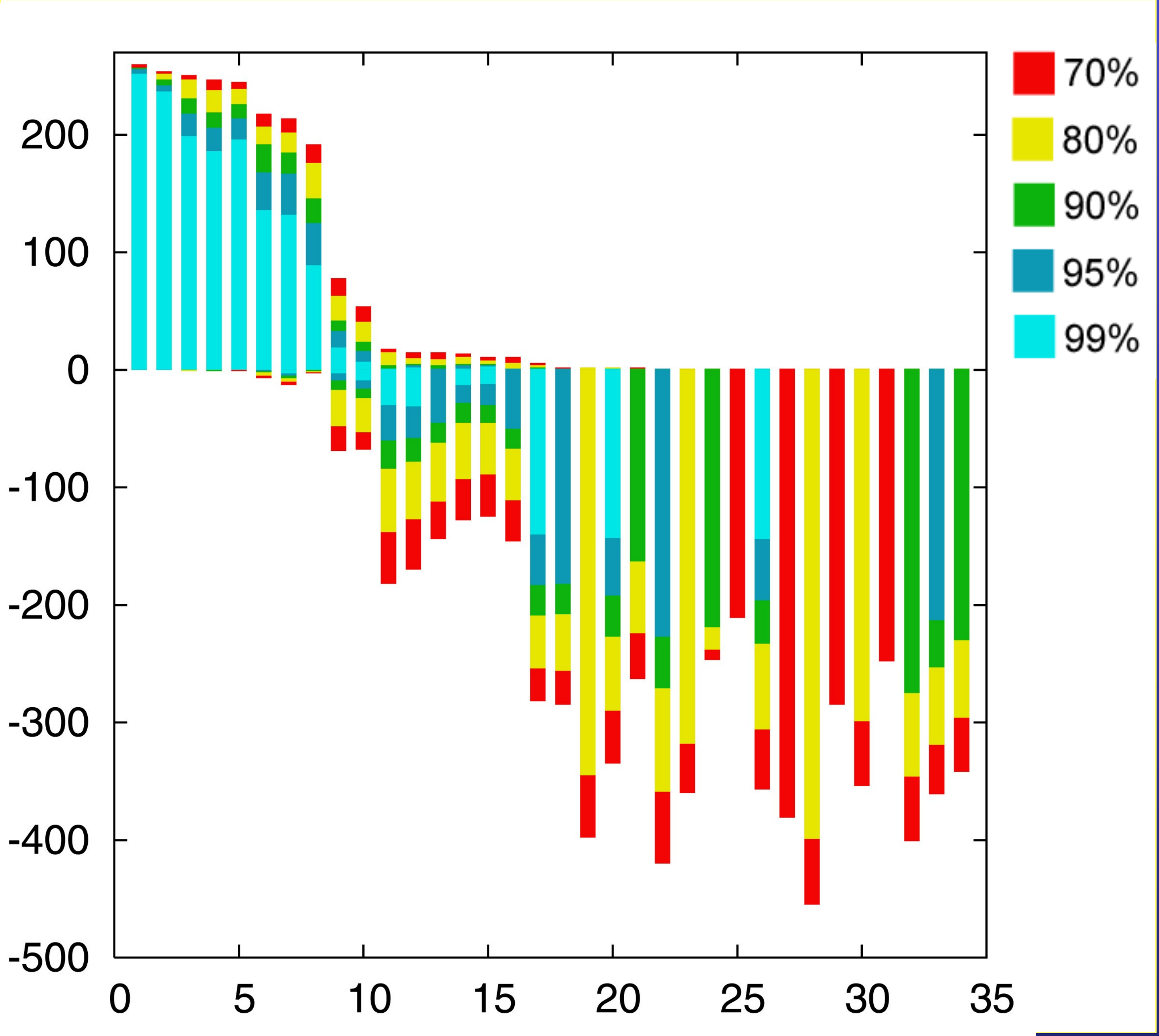
incompatible to illustrated bipartition

“Lento”-plot of 34 supported bipartitions (out of 4082 possible)

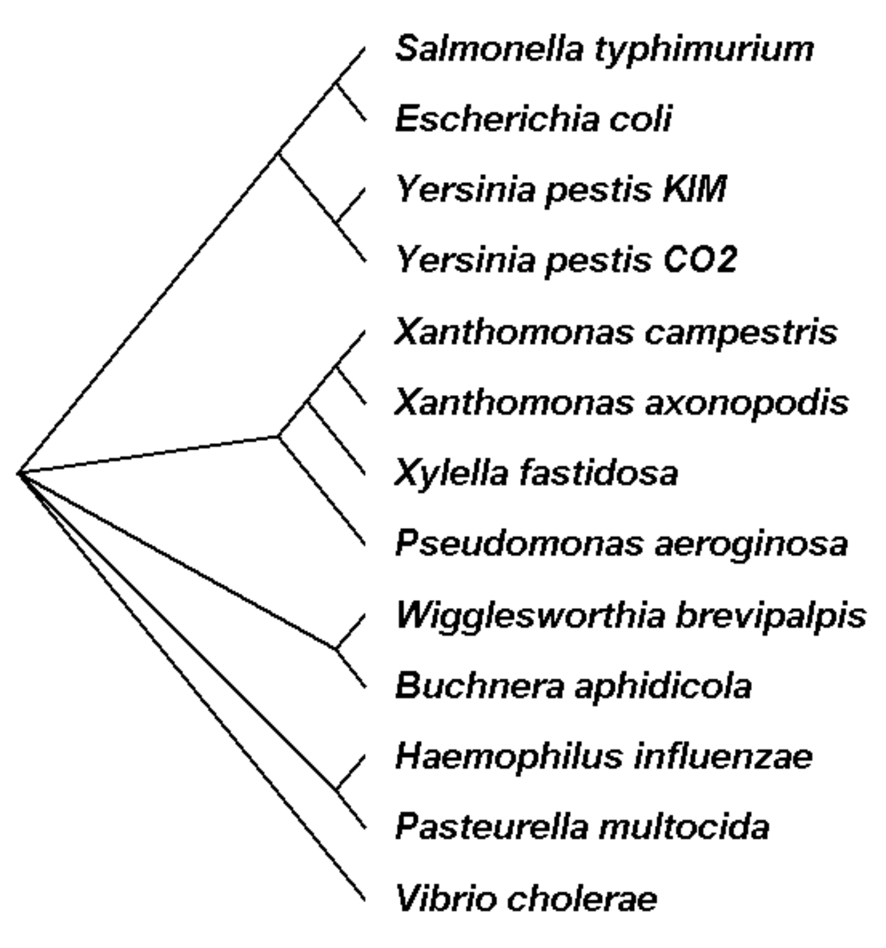
13 gamma-proteobacterial genomes (258 putative orthologs):

- E.coli
- Buchnera
- Haemophilus
- Pasteurella
- Salmonella
- Yersinia pestis (2 strains)
- Vibrio
- Xanthomonas (2 sp.)
- Pseudomonas
- Wigglesworthia

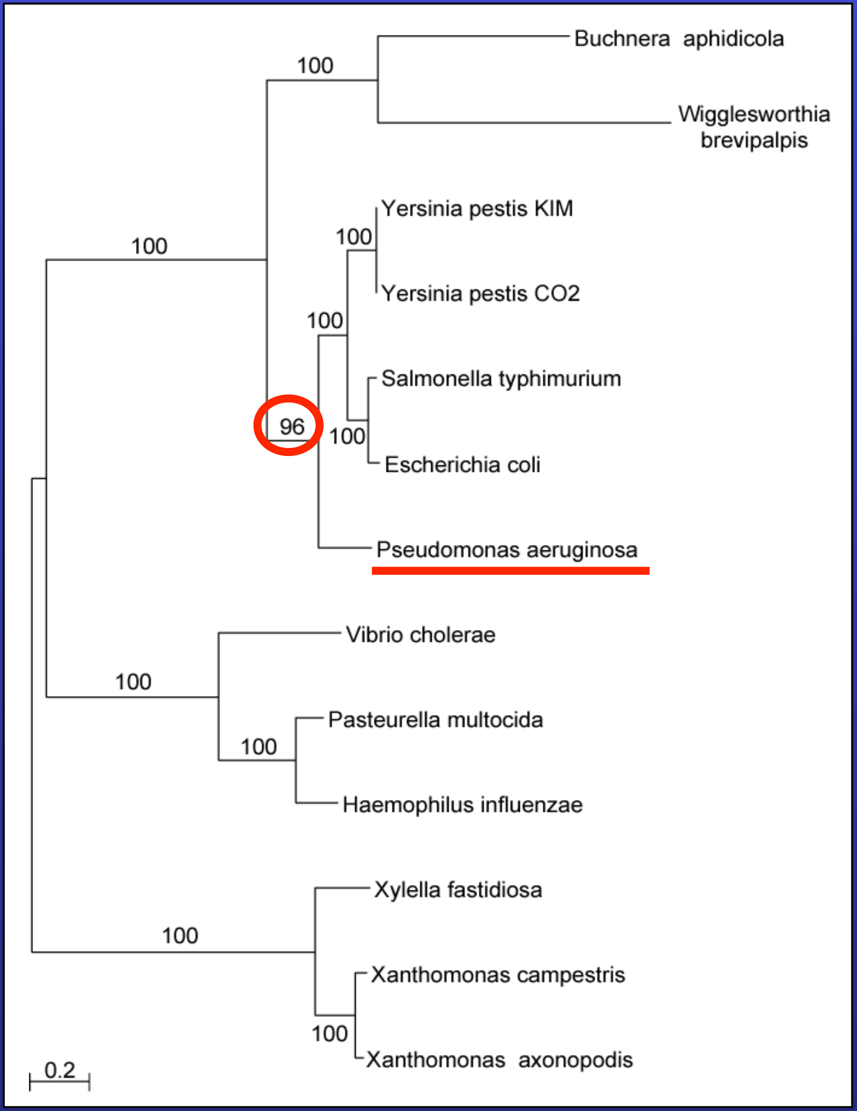
There are 13,749,310,575 possible unrooted tree topologies for 13 genomes



Consensus clusters of eight significantly supported bipartitions



Phylogeny of putatively transferred gene (virulence factor homologs (mviN))

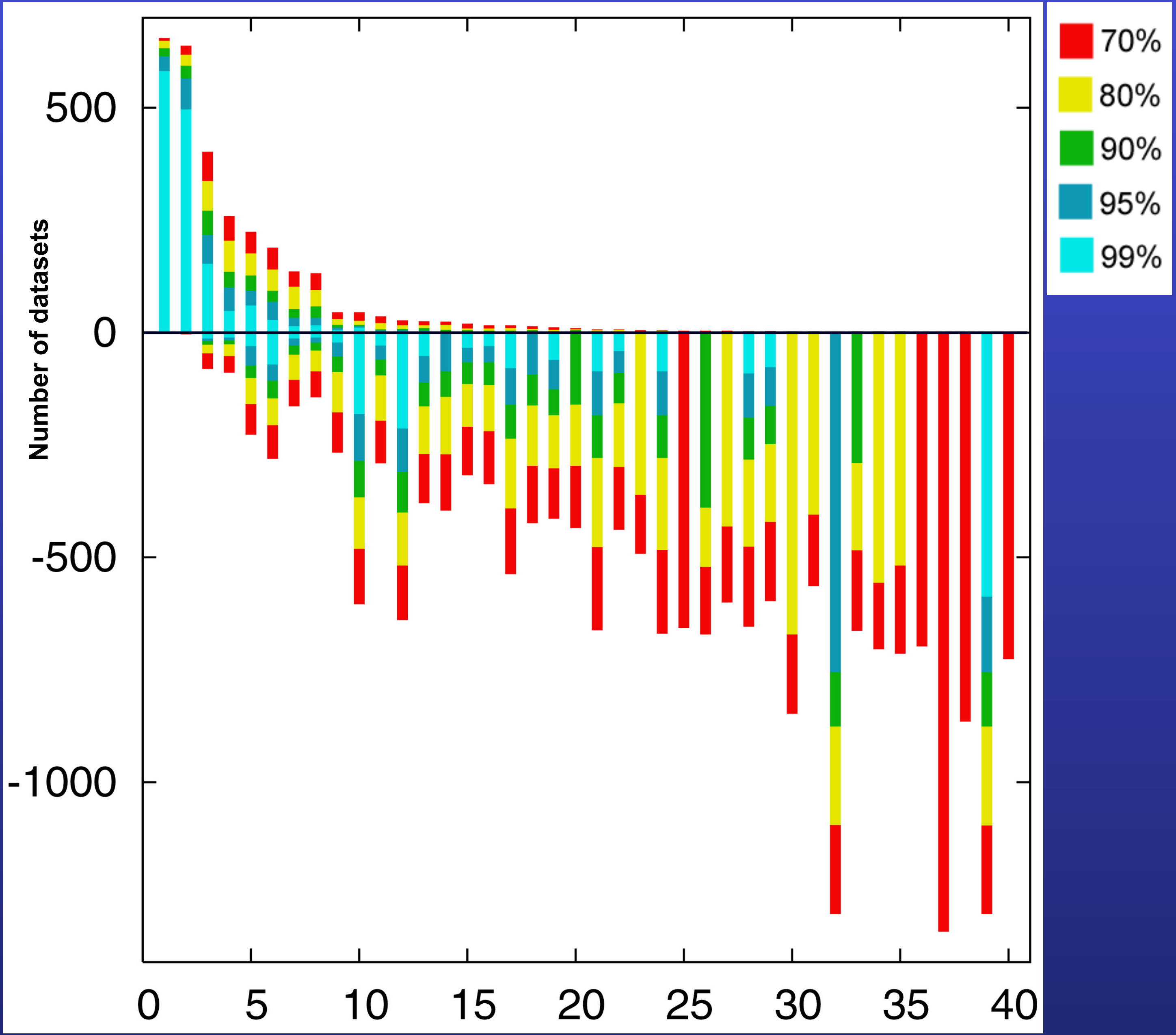


only 258 genes analyzed

“Lento”-plot of supported bipartitions (out of 501 possible)

10 cyanobacteria:

- *Anabaena*
- *Trichodesmium*
- *Synechocystis* sp.
- *Prochlorococcus marinus* (3 strains)
- Marine *Synechococcus*
- *Thermo-synechococcus elongatus*
- *Gloeobacter*
- *Nostoc punctioforme*

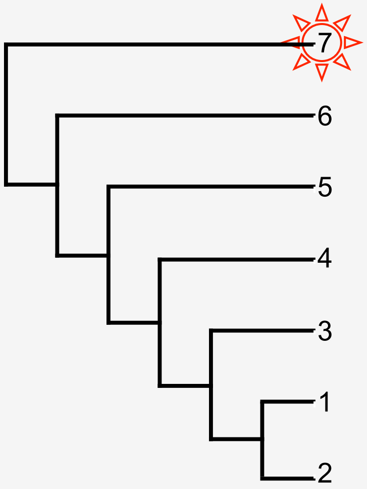


Based on 678 sets of orthologous genes

Zhaxybayeva, Lapierre and Gogarten, *Trends in Genetics*, 2004, **20(5)**: 254-260.

PROBLEMS WITH BIPARTITIONS (A)

Q_1={
4 5 6 7
1 5 6 7
2 5 6 7
3 5 6 7
3 4 6 7
1 4 6 7
2 4 6 7
2 3 6 7
1 3 6 7
1 2 6 7
1 2 3 7
1 2 4 7
1 3 4 7
2 3 4 7
2 3 5 7
1 3 5 7
1 2 5 7
1 4 5 7
2 4 5 7
3 4 5 7
3 4 5 6
1 4 5 6
2 4 5 6
2 3 5 6
1 3 5 6
1 2 5 6
1 2 3 6
1 2 4 6
1 3 4 6
2 3 4 6
2 3 4 5
1 3 4 5
1 2 4 5
1 2 3 5
1 2 3 4}



B_1={
**.....,
***....,
****...,
*****..
}

bipartitions

embedded quartets

A single rogue sequence that moves from one end of a Hennigian comb to the other changes all bipartition

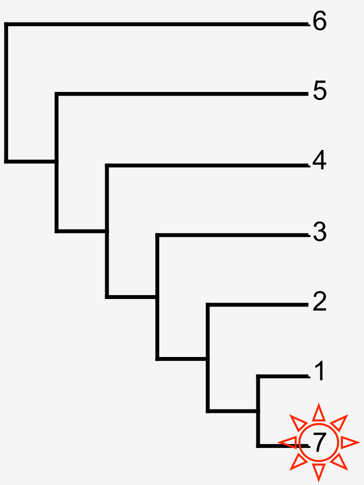
supported quartets

$Q1 \cap Q2 =$

{3 4 5 6, 1 4 5 6, 2 4 5 6, 2 3 5 6,
1 3 5 6, 1 2 5 6, 1 2 3 6, 1 2 4 6,
1 3 4 6, 2 3 4 6, 2 3 4 5, 1 3 4 5,
1 2 4 5, 1 2 3 5, 1 2 3 4}

supported bipartitions:

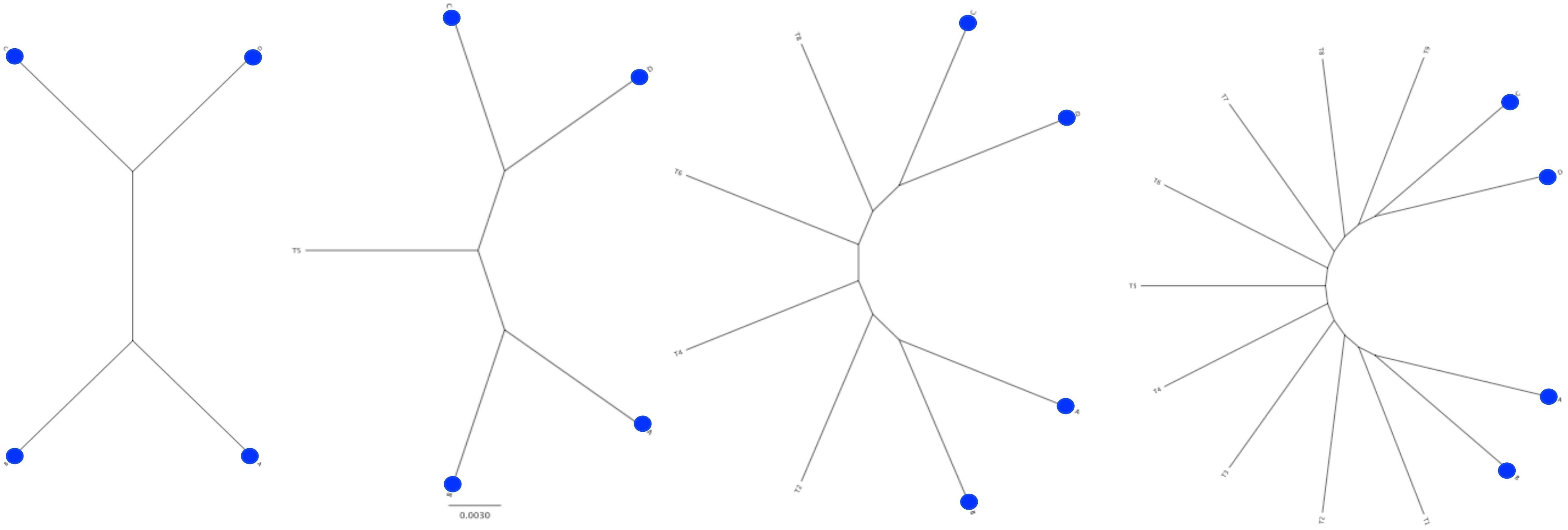
$B1 \cap B2 = \emptyset$



B_2={
.....,
**.....*,
***...*,
****..*
}

Q_2={
3 4 5 6
1 4 5 6
7 4 5 6
2 4 5 6
2 3 5 6
1 3 5 6
7 3 5 6
7 2 5 6
1 2 5 6
1 7 5 6
1 7 2 6
1 7 3 6
1 2 3 6
7 2 3 6
7 2 4 6
1 2 4 6
1 7 4 6
1 3 4 6
7 3 4 6
2 3 4 6
2 3 4 5
1 3 4 5
7 3 4 5
7 2 4 5
1 2 4 5
1 7 4 5
1 7 2 5
1 7 3 5
1 2 3 5
7 2 3 5
7 2 3 4
1 2 3 4
1 7 3 4
1 7 2 4
1 7 2 3}

Decay of bipartition support with number of OTUs



Phylogenies used for simulation

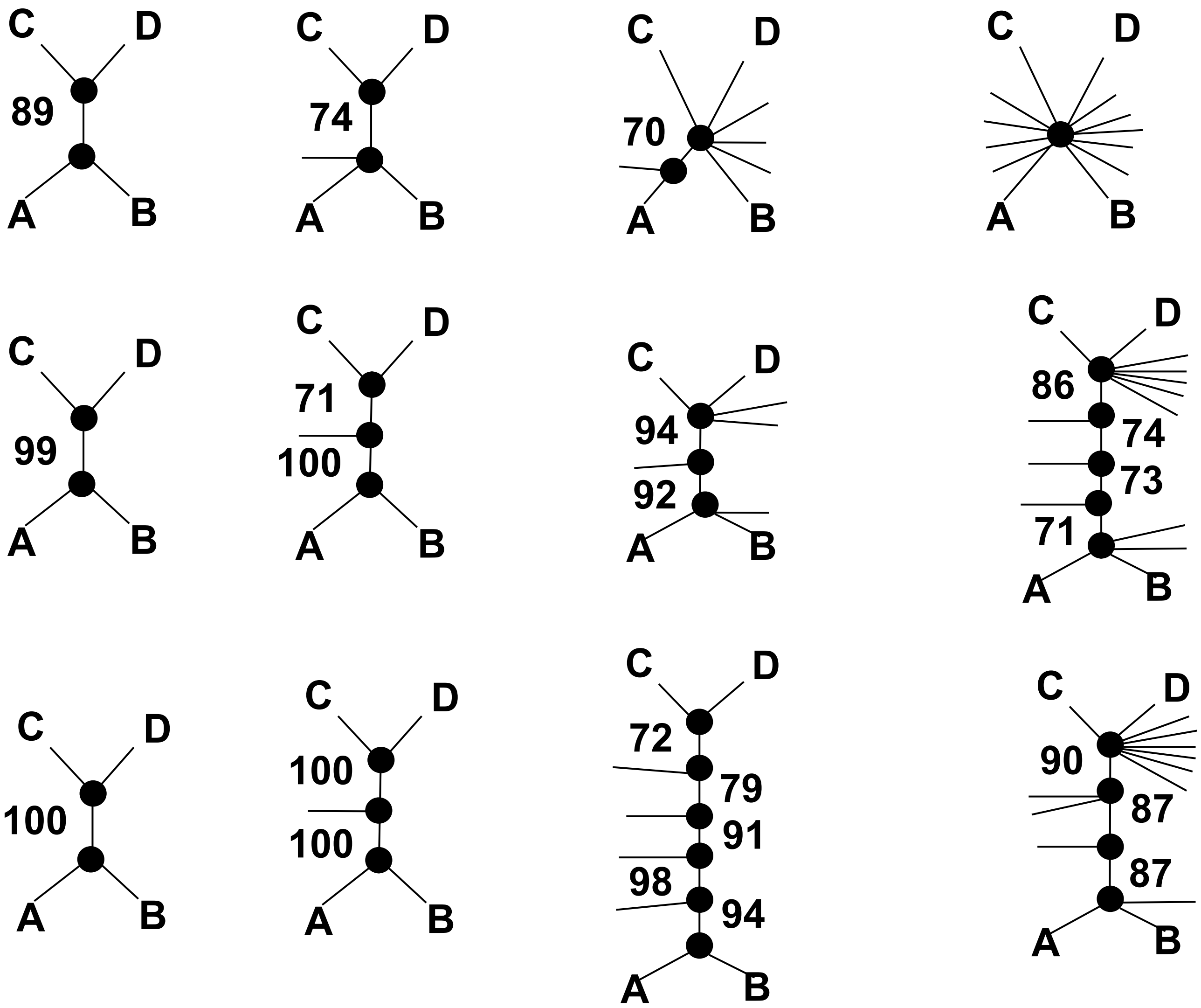
Example for decay of bipartition support with number of OTUs

Sequence lengths

200

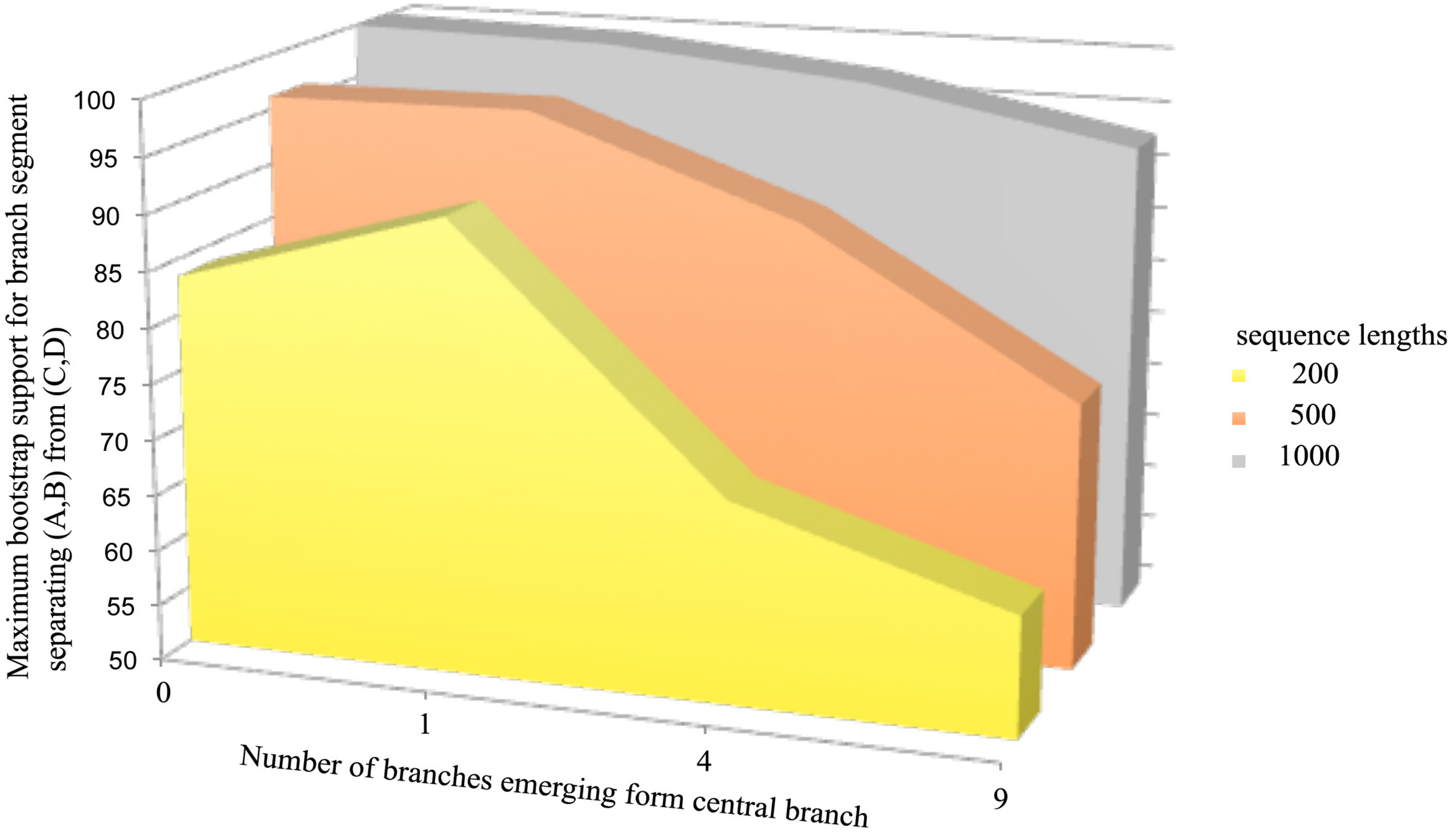
500

1000



Only branches with better than 70% bootstrap support are shown

Decay of bipartition support with number of OTUs

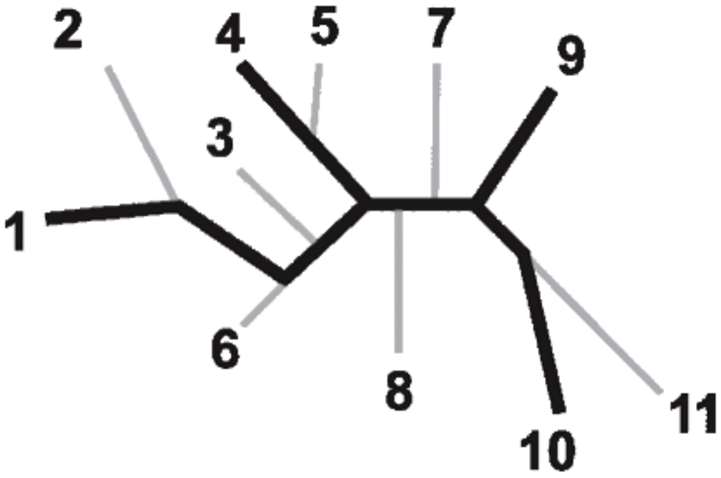


Each value is the average of 10 simulations using seq-gen.
Simulated sequences were evaluated using PHYML.
Model for simulation and evaluation WAG + Γ(α=1, 4 rate categories)

Bipartition Paradox:

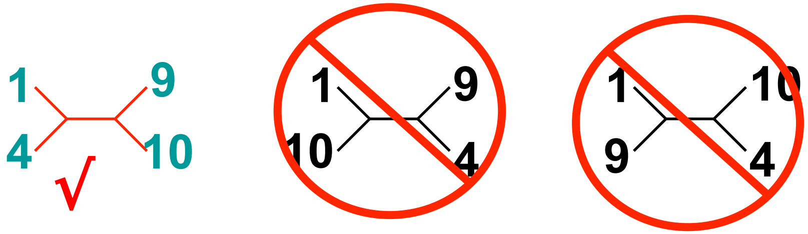
- The more sequences are added, the lower the support for bipartitions that include all sequences. The more data one uses, the lower the bootstrap support values become.
- This paradox disappears when only embedded splits for 4 sequences are considered.

Bootstrap support values for embedded quartets



— + — : tree calculated from one pseudo-sample generated by bootstraping from an alignment of one gene family present in 11 genomes

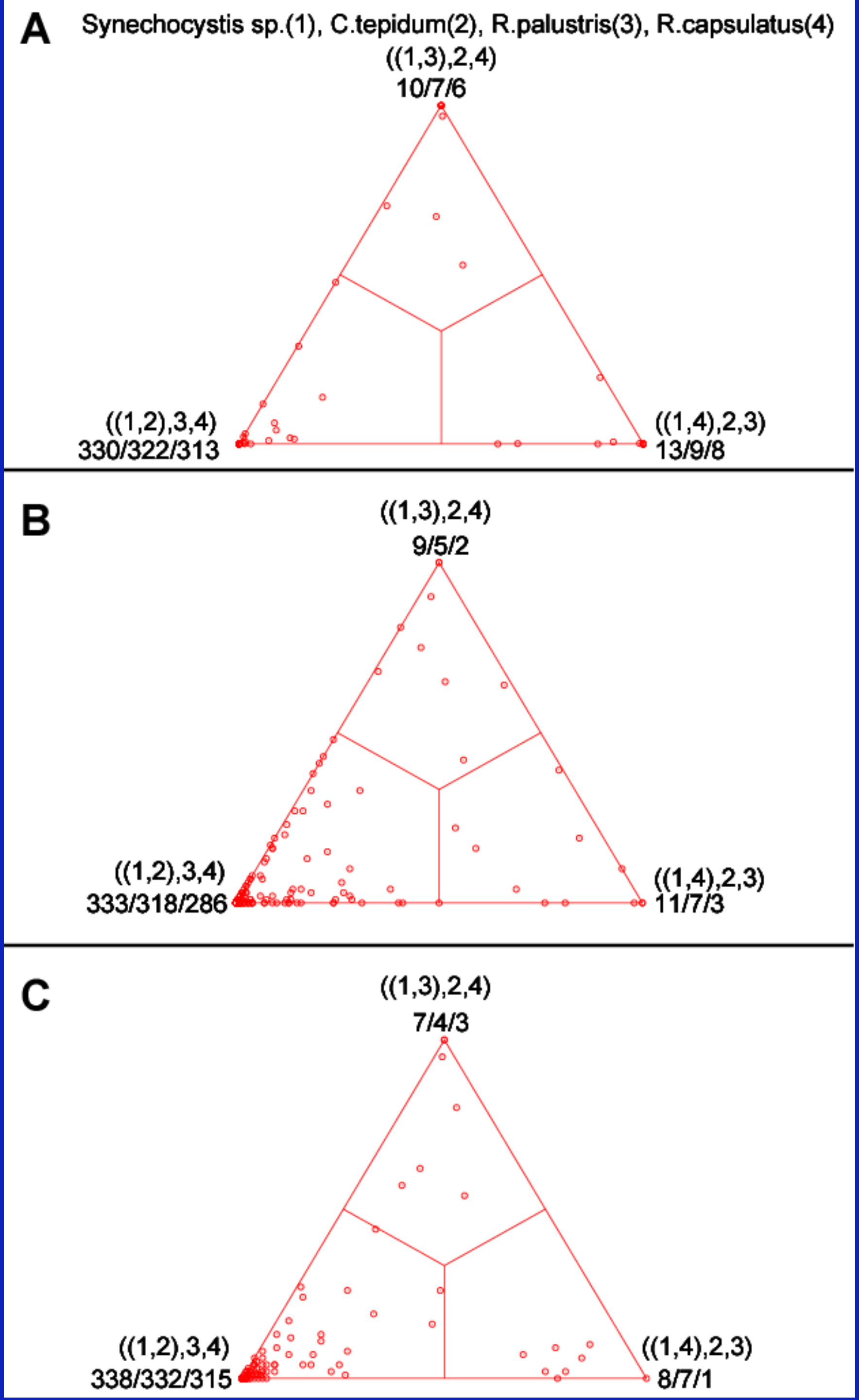
— : embedded quartet for genomes 1, 4, 9, and 10 .
This bootstrap sample supports the topology ((1,4),9,10).



Quartet spectral analyses of genomes iterates over three loops:

- ➢Repeat for all bootstrap samples.
- ➢Repeat for all possible embedded quartets.
- ➢Repeat for all gene families.

Zhaxybayeva et al. 2006, Genome Research, 16(9):1099-108



Zhaxybayeva and Gogarten, *BMC Genomics* 2003 4: 37

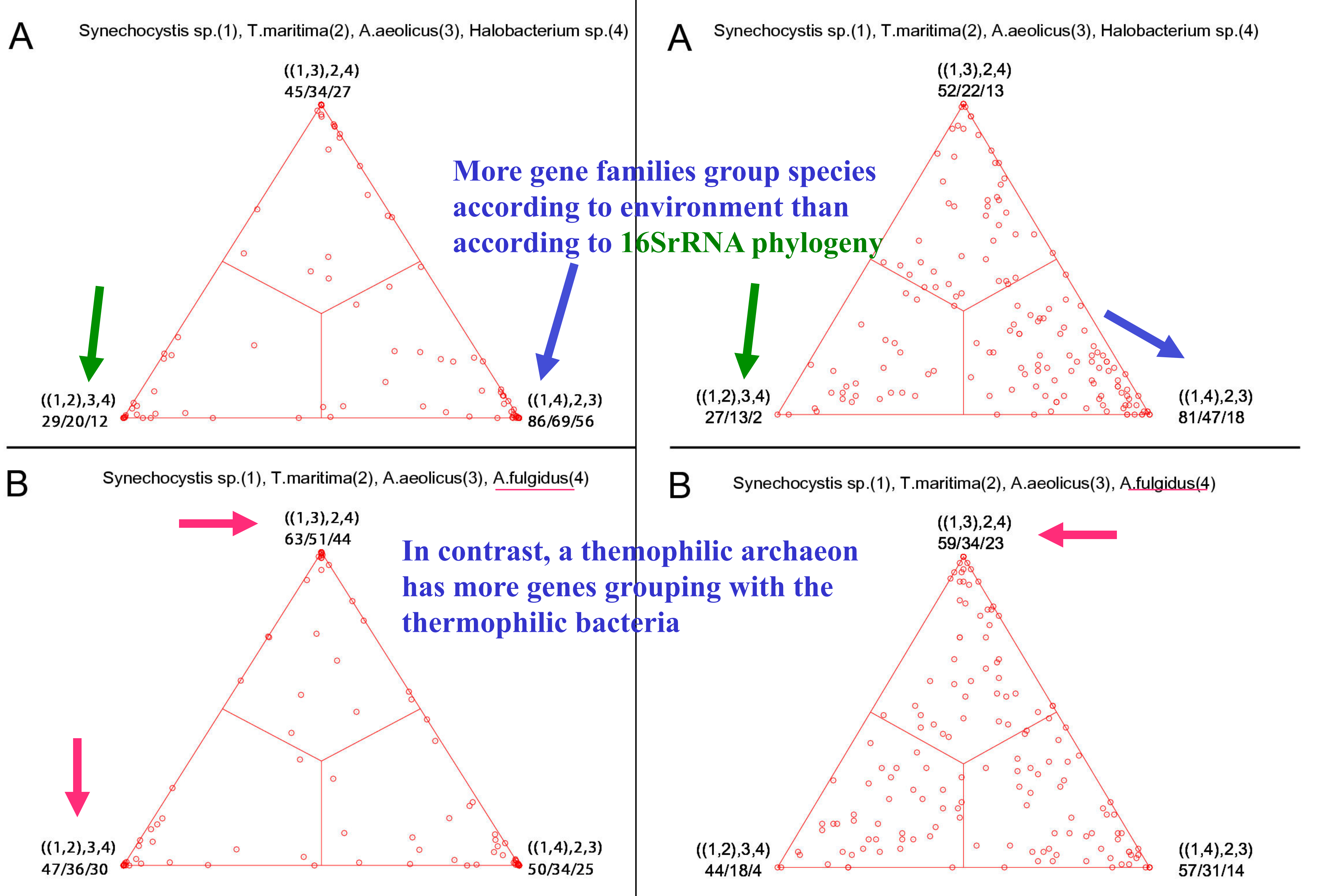
COMPARISON OF DIFFERENT SUPPORT MEASURES

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

B: mapping of bootstrap support values

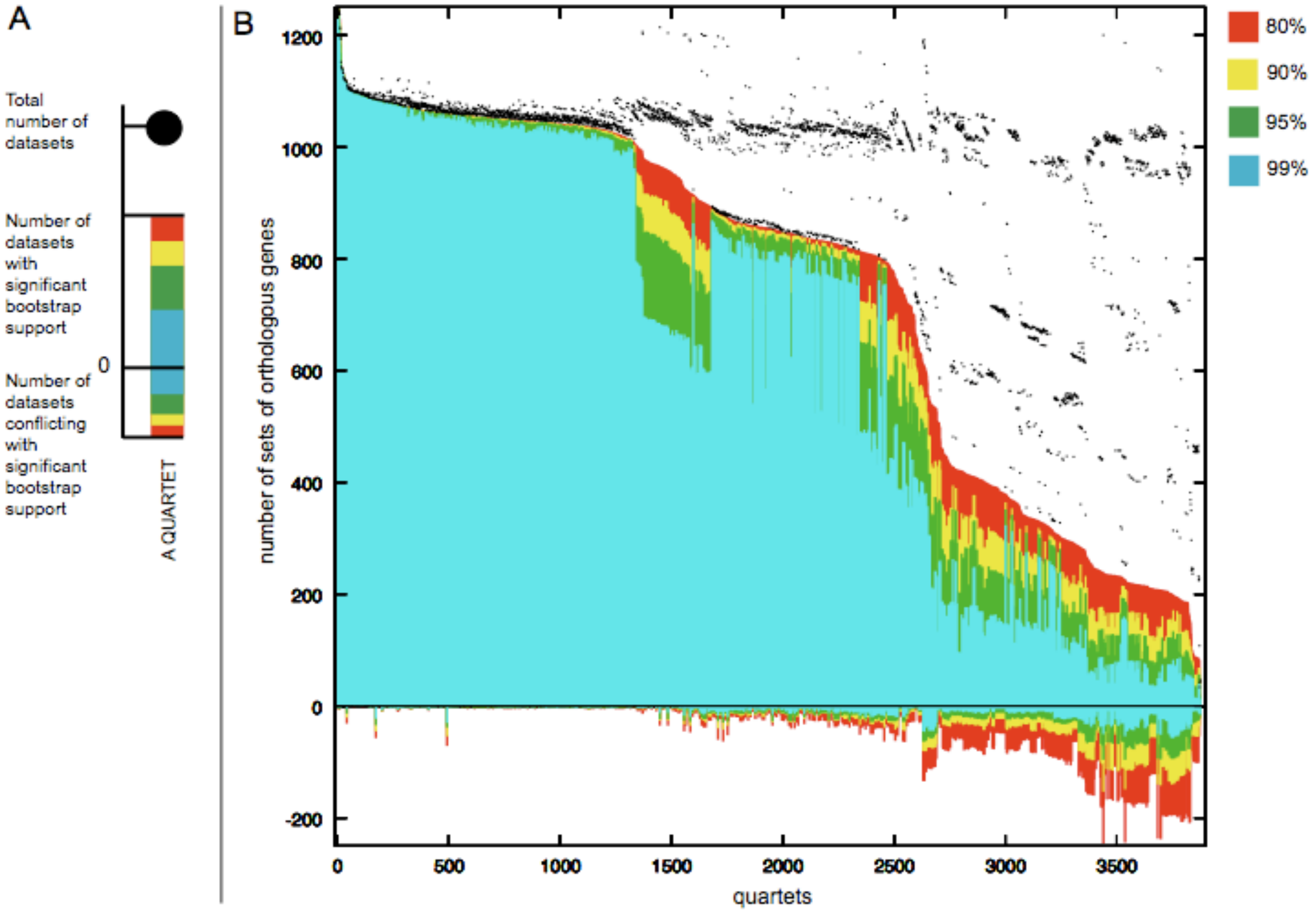
C: mapping of bootstrap support values from extended datasets

ml-mapping *versus* bootstrap values from extended datasets



More gene families group species according to environment than according to 16SrRNA phylogeny

In contrast, a themophilic archaeon has more genes grouping with the thermophilic bacteria



Quartet decomposition analysis of 19 *Prochlorococcus* and marine *Synechococcus* genomes. Quartets with a very short internal branch or very long external branches as well those resolved by less than 30% of gene families were excluded from the analyses to minimize artifacts of phylogenetic reconstruction.