

Bio334 – Building Maximum Likelihood Trees

Solutions to the exercises
Janko Tackmann and Lukas Malfertheiner

For remaining doubts feel free to contact us on Slack or at:
Janko.tackmann@uzh.ch or Lukas.malfertheiner@uzh.ch

Exercise 1 – Part 1: The grammar

Is the tree specified as rooted or unrooted?

$((A,(B,(C,D))), (E,F));$

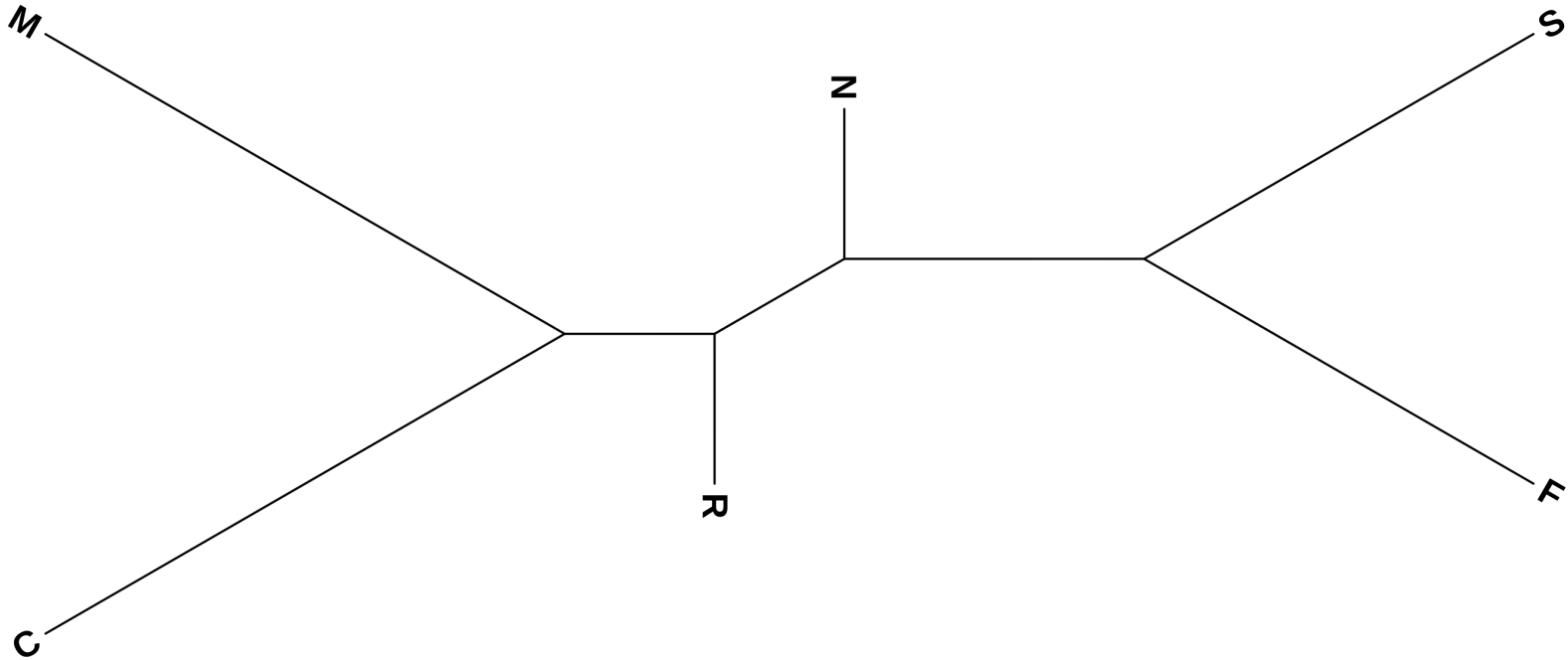
Exercise 1 – Part 1: The grammar

Is the tree specified as rooted or unrooted?

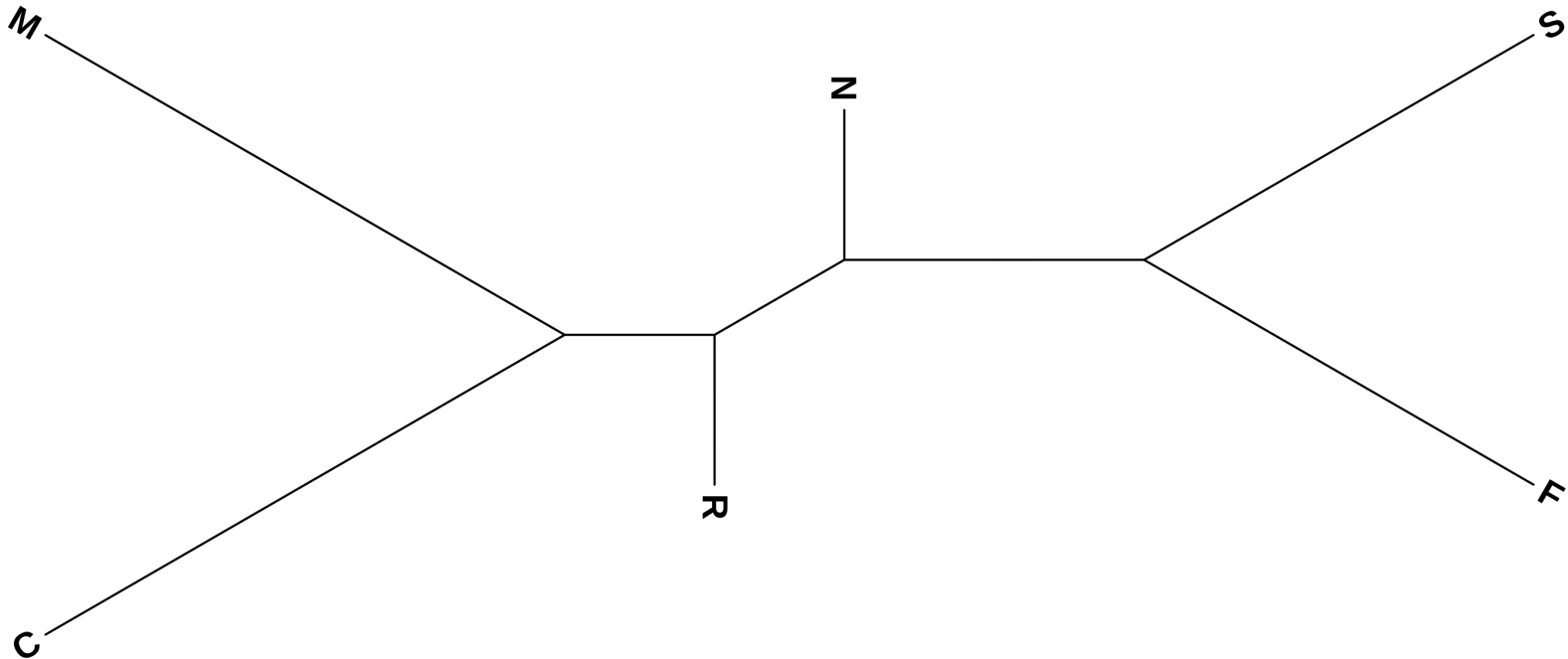
$((A, (B, (C, D))), (E, F));$

-> The tree is rooted

Exercise 1 – Part 2: Specifying branch length

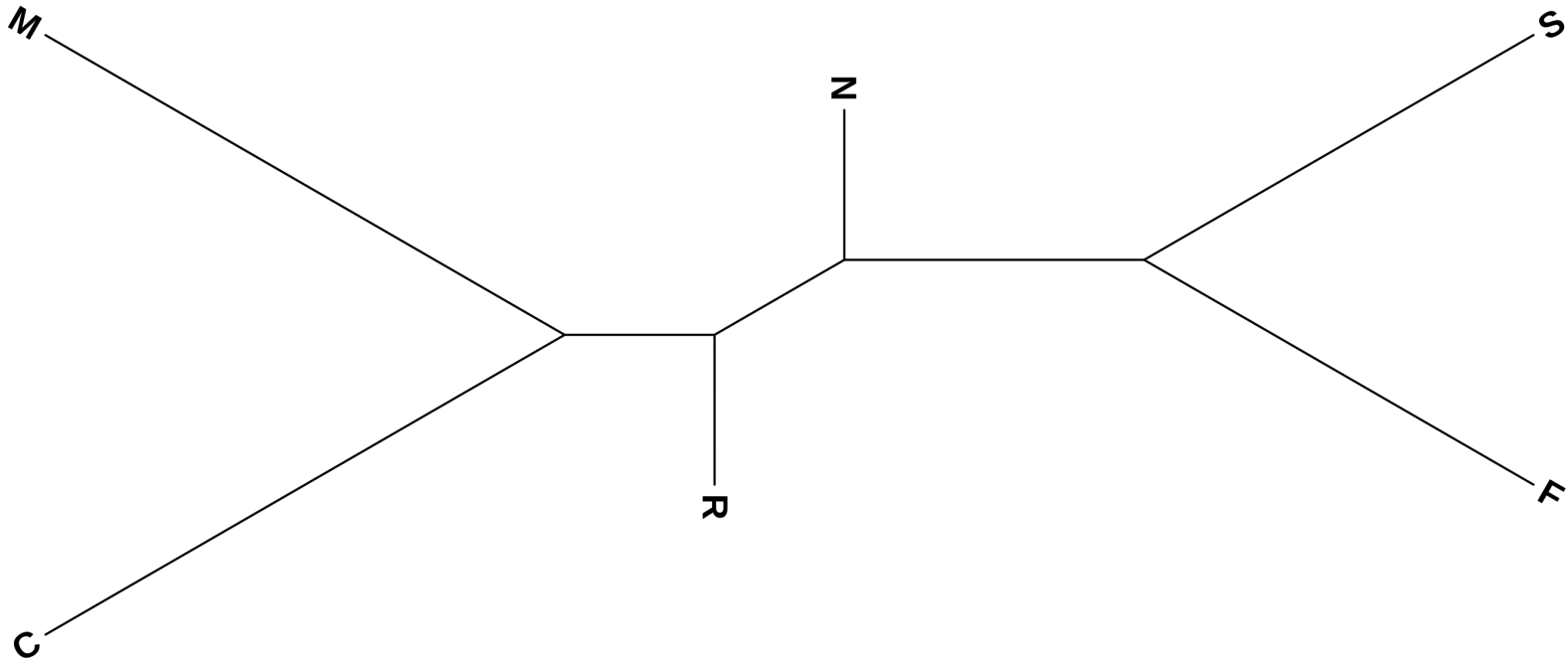


Exercise 1 – Part 2: Specifying branch length



$(N, (R, (C, M)), (S, F));$

Exercise 1 – Part 2: Specifying branch length



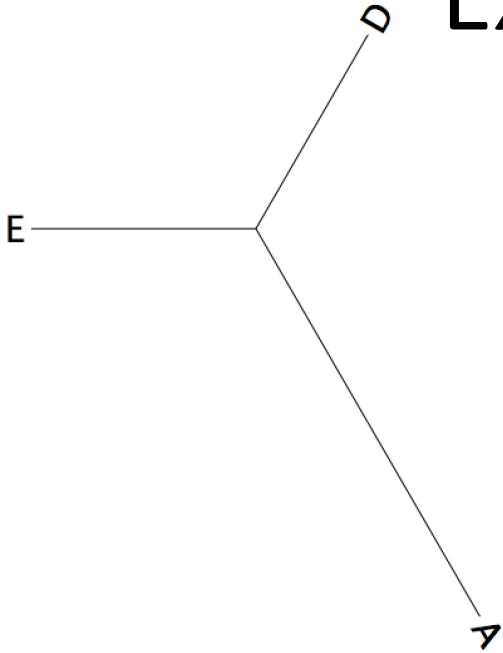
(N:0.5,(R:0.5,(C:2,M:2):0.5):0.5,(S:1.5,F:1.5):1);

Exercise 1 – Part 3: Common errors in Newick representations

Exercise 1 – Part 3.a

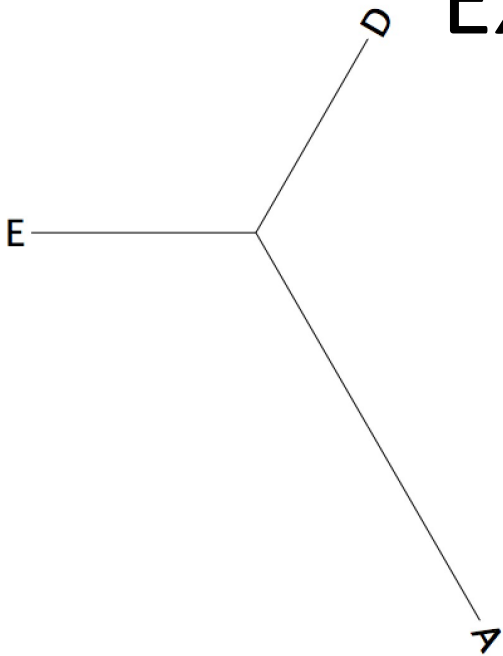
$(A, (E, D)), (C, (B, F));$

Exercise 1 – Part 3.a

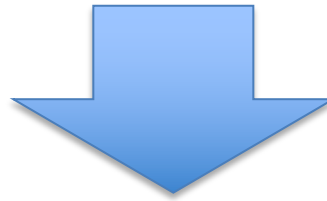


$(A, (E, D)), (C, (B, F));$

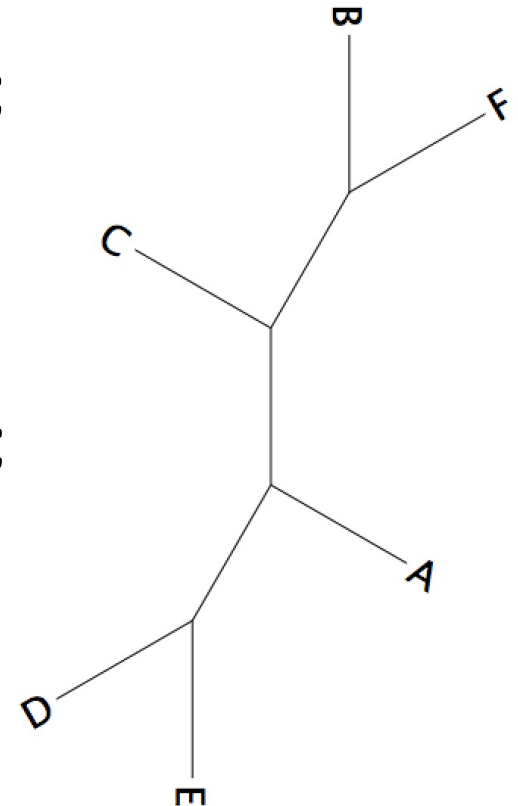
Exercise 1 – Part 3.a



$(A,(E,D)),(C,(B,F));$



$(A,(E,D),(C,(B,F)))$

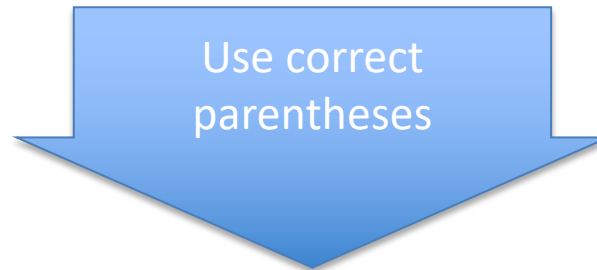


Exercise 1 – Part 3.b

(A:1,D:6,([E:1.01,F:1.2]:1,B:2):0.21,(C:4,G:2.2):2):1);

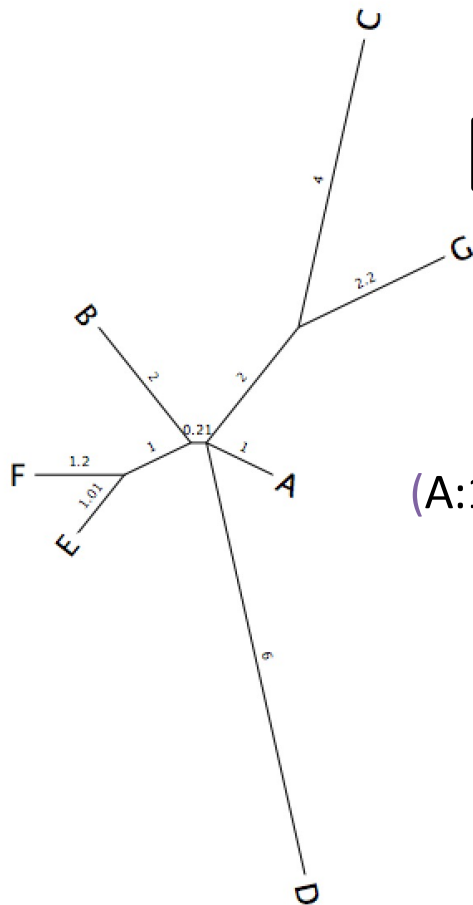
Exercise 1 – Part 3.b

(A:1,D:6,([E:1.01,F:1.2]:1,B:2):0.21,(C:4,G:2.2):2):1);



(A:1,(D:6,((E:1.01,F:1.2):1,B:2):0.21),(C:4,G:2.2):2):1);

Exercise 1 – Part 3.b

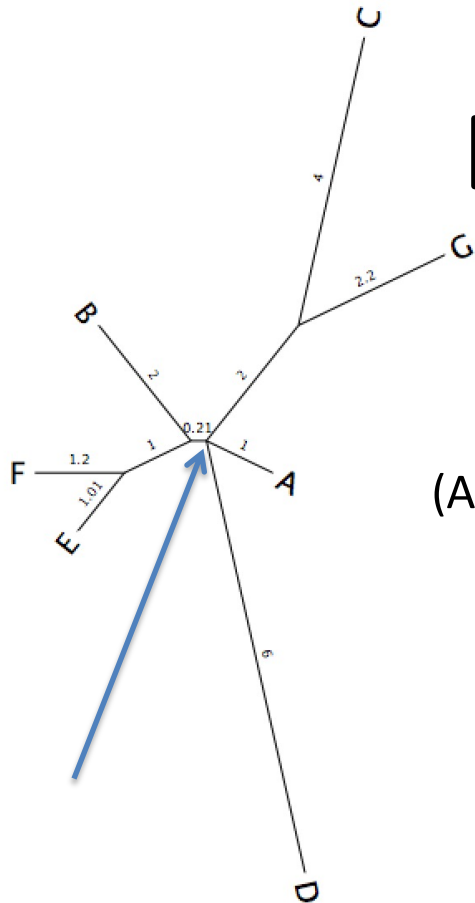


(A:1,D:6,((E:1.01,F:1.2):1,B:2):0.21,(C:4,G:2.2):2):1);



(A:1,(D:6,((E:1.01,F:1.2):1,B:2):0.21),(C:4,G:2.2):2);

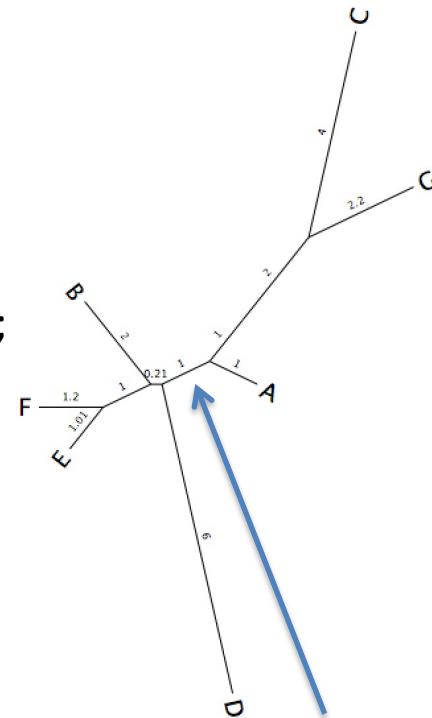
Exercise 1 – Part 3.b



(A:1,D:6,((E:1.01,F:1.2):1,B:2):0.21,(C:4,G:2.2):2);

Resolve multi-branching point

(A:1,(D:6,((E:1.01,F:1.2):1,B:2):0.21):1,(C:4,G:2.2):2);



Note: There are many ways to solve the multifurcation, this is just one of the many.

Exercise 2 – Overview

Small theory recap

Phylogenetic trees are used to represent the evolutionary relationships between a group of related sequences/species/genes. For tree construction, several computational methods are available. These are as follows:

1. **Distance Based Method:** Neighbor Joining, etc. These require a distance measure between the sequences.
2. **Maximum Parsimony:** The tree based on this method will provide the minimum number of evolutionary steps to produce the sequences.
3. **Maximum Likelihood:** This method uses an expected pattern of mutational changes from one DNA base to another with probability calculations to find the most likely arrangement of branches that generates the set of sequences.

$$L = p(\text{data} \mid \text{tree, branch lengths, model})$$

The ML algorithm searches different trees and branch lengths to find the L_{\max} . It works as following

LOOP OVER

Generate tree topology → Optimize branch lengths → Retain if result improved

Exercise 2 – Part 1.d

Can you identify which command line arguments are required (not-optional) for RAxML to be executed?

- s alignment_file_name
- n output_file_name_extension
- m model_of_amino_acid_substitution
- p random_seed

Exercise 2 – Part 2

Using RAxML on the MFS-1 dataset

raxmlHPC

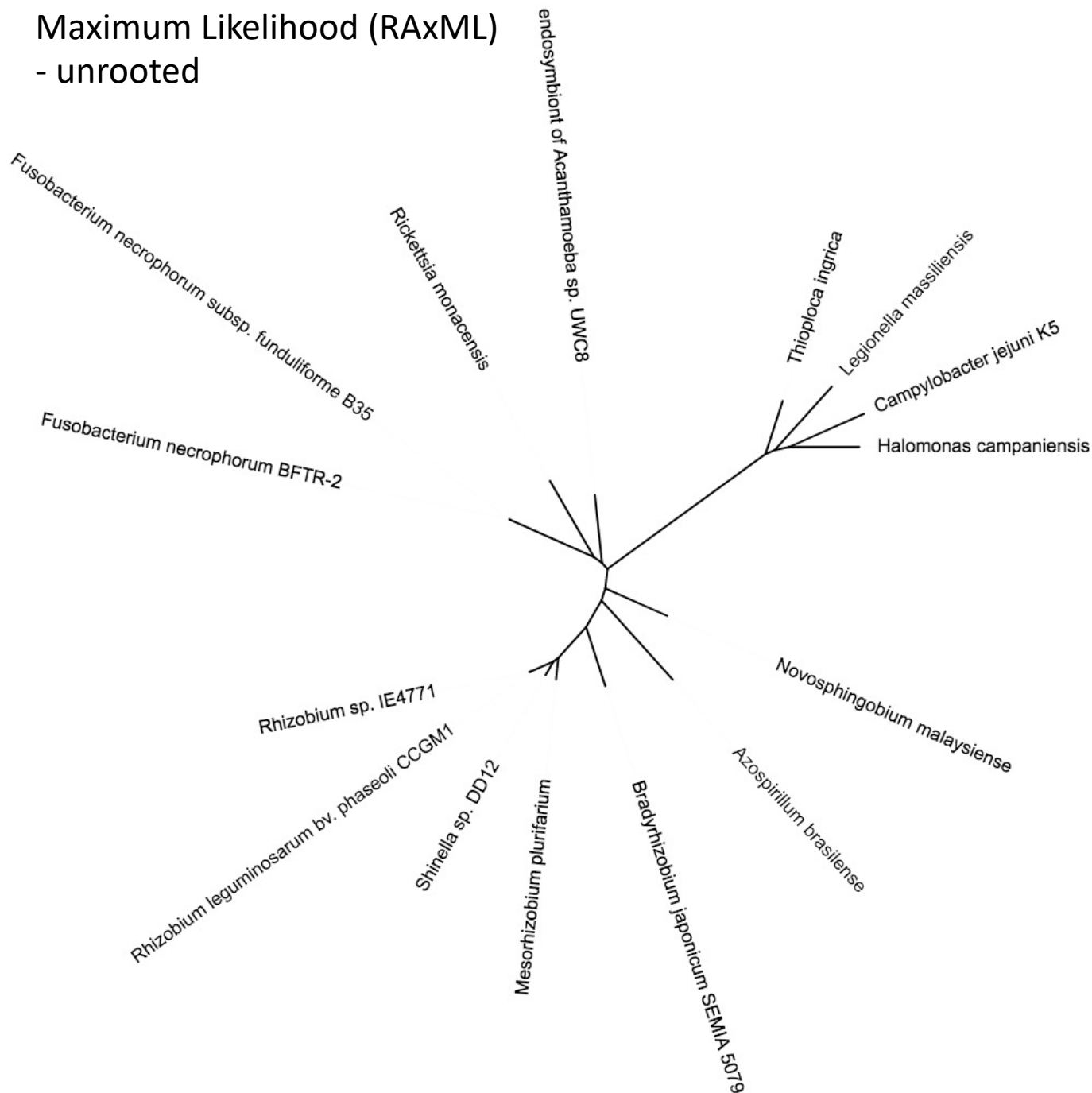
-s mfs_domain_proteins_aligned_taxnames.fa

-n msf.auto.txt

-m PROTGAMMAAUTO

-p 123

Maximum Likelihood (RAxML) - unrooted



Bacteria

Fusobacteria

Fusobacteriia

Fusobacteriales

Fusobacteriaceae

Fusobacterium

Proteobacteria

Gammaproteobacteria

Thioploca

Halomonas

Legionella

Campylobacter

Alphaproteobacteria

Novosphingobium

Rickettsia

Rhizobiales

Mesorhizobium

Rhizobiaceae

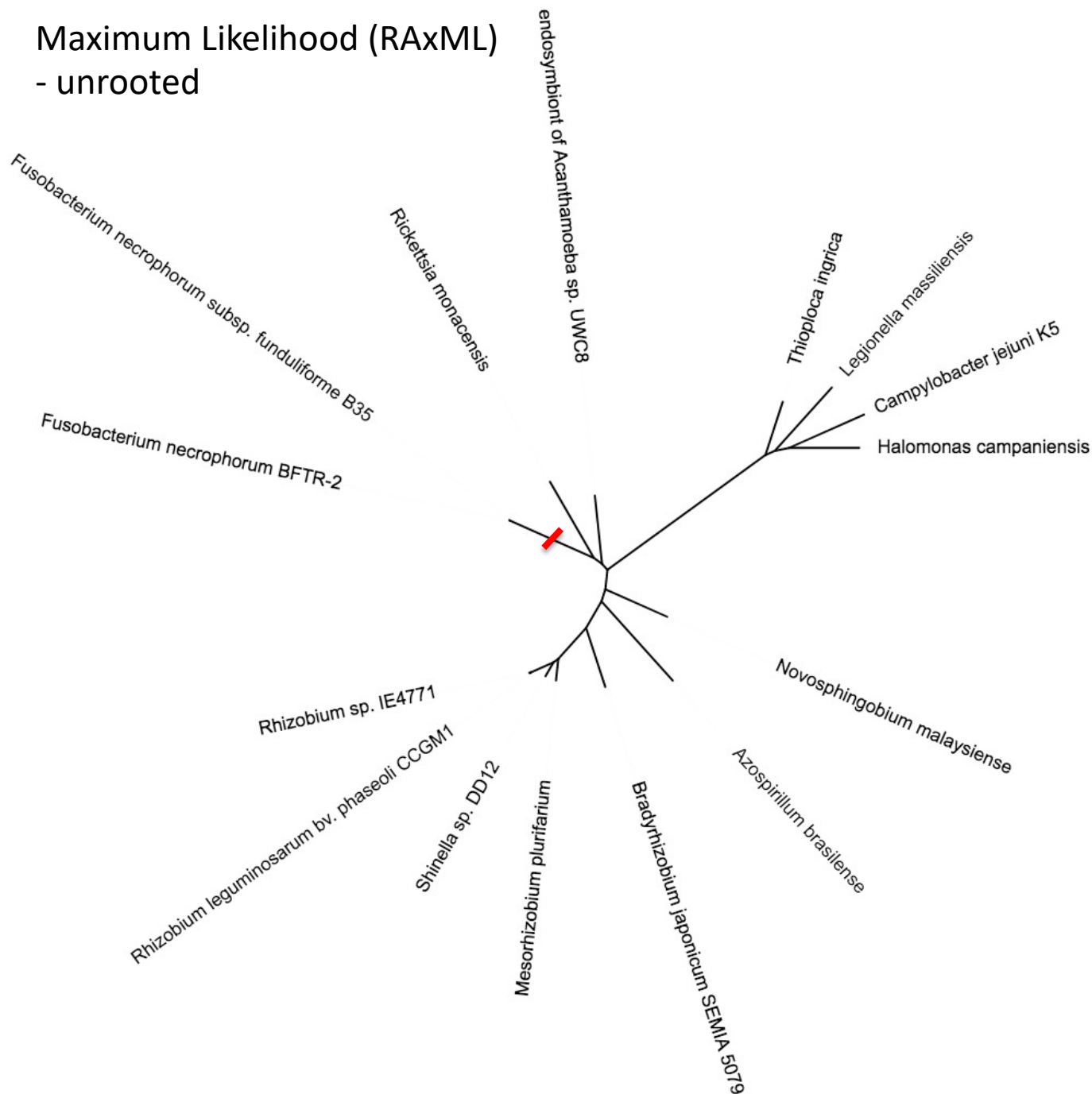
Shinella

Rhizobium

Bradyrhizobium

Azospirillum

Maximum Likelihood (RAxML) - unrooted



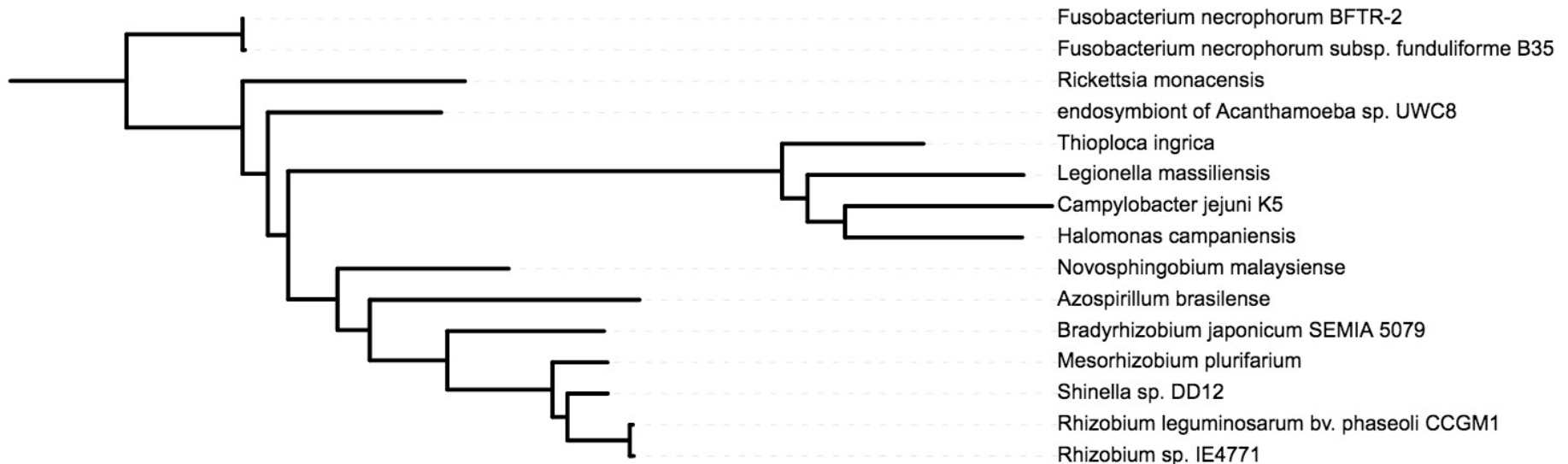
Maximum Likelihood (RAxML, rooted on Fusobacteria)



Neighbor joining (rooted on Fusobacteria)



Maximum Likelihood (RAxML, rooted on Fusobacteria)



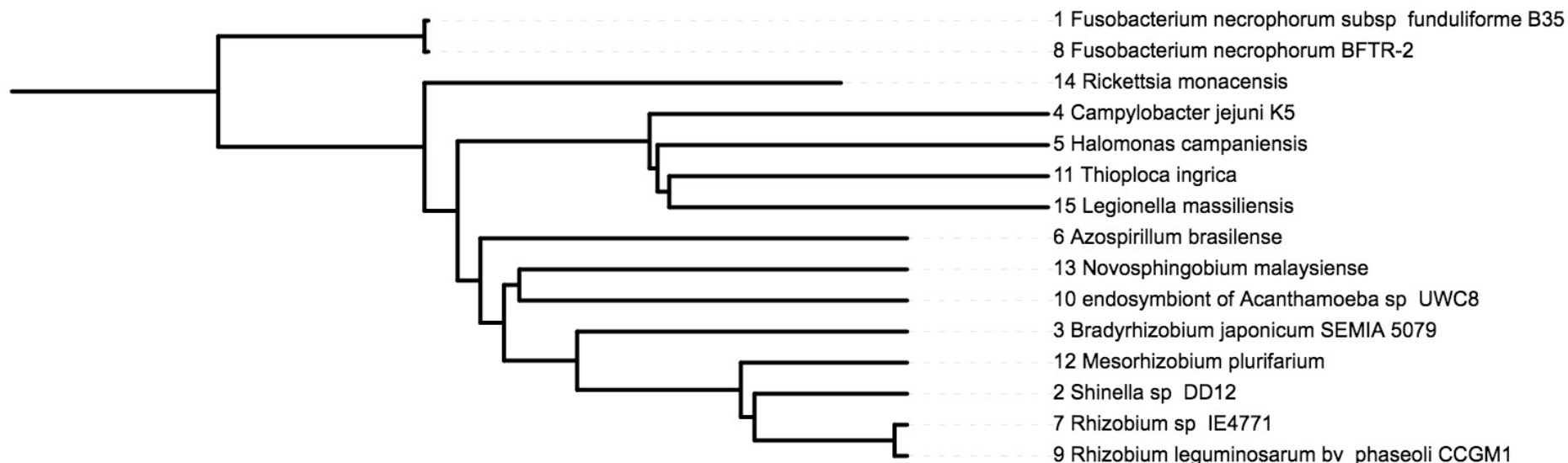
Neighbor joining (rooted on Fusobacteria)



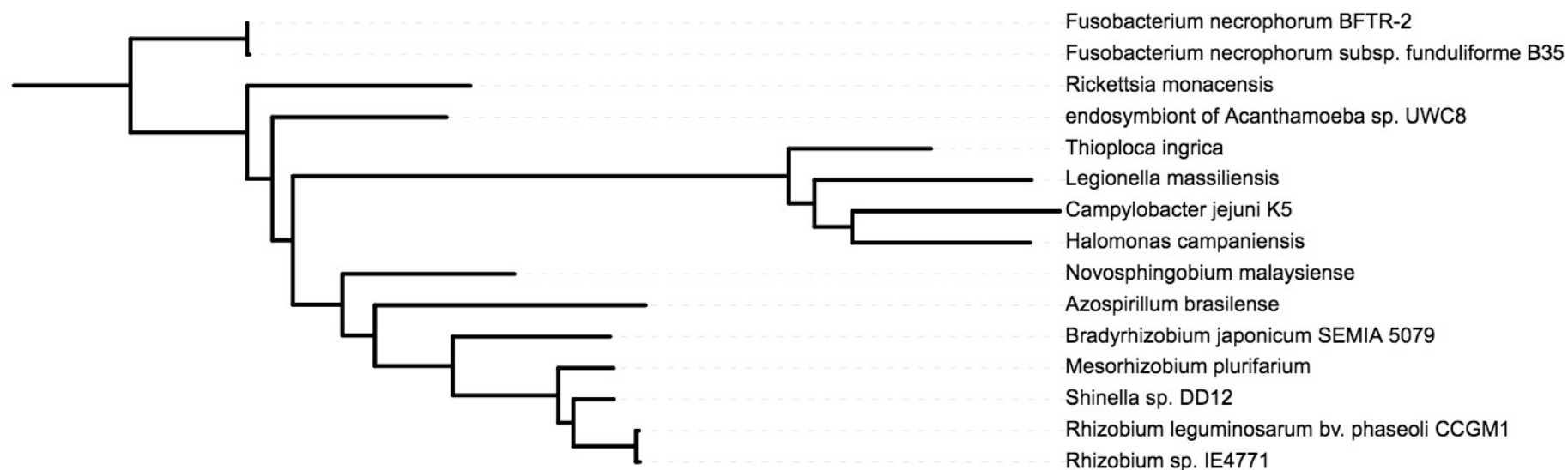
Maximum Likelihood (RAxML, rooted on Fusobacteria)



UPGMA (rooted on Fusobacteria)



Maximum Likelihood (RAxML, rooted on Fusobacteria)



Exercise 3 – Part 1

Sequence alignment

```
HTLV      SHRFKNLGAQTGELWNTFLKTAAPLAPVKALMPVFTLSPVIINTAPCLFSDGSTSR----
HIV1B5    -----WE-FVNT-PPLVKL-----WYQLEKEPIVGAETFFYVDGAASRETKL
HIV1H2    -----WE-FVNT-PPLVKL-----WYQLEKEPIVGAETFFYVDGAANRETKL
HIV1PV    -----WE-FVNT-PPLVKL-----WYQLEKEPIVGAETFFYVDGAANRETRL
HIV1N5    -----WE-FVNT-PPLVKL-----WYQLEKEPIIGAETFFYVDGAANRETKL
HIV1OY    -----WE-FVNT-PPLVKL-----WYQLEKDIIGAETFFYVDGAANRETKL
HIV1ND    -----WE-FVNT-PPLVKL-----WYQLEKEPIIGAETFFYVDGAANRETKL
HIV1Z2    -----WE-FVNT-PPLVKL-----WYQLEKEPIIGAETFFYVDGAANRETKL
HIV1MN    -----WE-FVNT-PPLVKL-----WYQLEKEPIVGAETFFYVDGAANRETKK
HIV1U4    -----WE-FVNT-PPLVKL-----WYQLEKDIIGAETFFYVDGAANRETKL
SIVCZ     -----WE-FINT-PPLVKL-----WYSLETEPIPTTDTYYVDGAANRETKT
HIV2CA    -----WD-FVST-PPLVRL-----AFNLVGDPPIPGTETFFYTDGSCNRQSKE
HIV2RO    -----WD-FVST-PPLVRL-----AFNLVGDPPIGAETFFYTDGSCNRQSKE
HIV2SB    -----WD-FVST-PPLVRL-----AFNLVKDPIPGAETFFYTDGSCNRQSKE
HIV2KR    -----WD-FVST-PPLVRL-----AFNLVKDPIGGEETFFYTDGSCNRQSKE
HIV2ST    -----WD-FIST-PPLVRL-----VFNLVKDPIIGAETFFYTDGSCNKQSRE
HIV2D1    -----WD-FVST-PPLVRL-----TFNLVGDPPIPGTETFFYTDGSCNRQSKE
HIV2G1    -----WD-FVST-PPLVRL-----TFNLVGDPPIGAETFFYTDGSCNRQSKE
Smanga_S4 -----WD-FVST-PPLVRL-----VFNLVKEPIQGAETFFYVDGSCNRQSRE
Smanga_SP -----WD-FVST-PPLVRL-----VFNLVKEPIQGAETFFYVDGSCNRQSRE

* : . : * . * . :      : *      *      : * * : . :
```

```
HTLV      -----LANTGASR-----
HIV1B5    DRQ-GTVSFNFPQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGG
HIV1H2    DRQ-GTVSFNFPQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGG
HIV1PV    DRQ-GTVSFNFPQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGG
HIV1N5    DRQ-GTVSFNFPQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGG
HIV1OY    DRQ-GTVSFNFPQITLWQRPIVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGG
HIV1ND    ERQ-GTVSFNFPQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGG
HIV1Z2    ERQ-GTVSFNFPQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGG
HIV1MN    DRQ-GTVSFNFPQITLWQRPIVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGG
HIV1U4    ERQ-GTVSFNFPQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGG
SIVCZ     -RE-QSISTNLPQITLWQRPLIPVKVEGQLCEALLDTGADDTVIERIQLQGLWKPKMIGG
HIV2CA    MQG-DNRGLAAPQFSLWKRPPVTAHIEGQFVEVLLDTGADDSIVAGIELGSNYSPKIVGG
HIV2RO    IQGATNRLGLAAPQFSLWKRPPVTAYIEGQFVEVLLDTGADDSIVAGIELGSNYSPKIVGG
HIV2SB    TQR-DDRGLAAPQFSLWKRPPVTAYIEDQFVEVLLDTGADDSIVAGIELGSNYSPKIVGG
HIV2KR    TQR-GDRGLAAPQFSLWKRPPVTAYIEGQFVEVLLDTGADDSIVAGIELGSNYSPKIVGG
HIV2ST    MQR-DDRGLAAPQFSLWKRPPVTAYIEGQFVEVLLDTGADDSIVAGIELGSNYSPKIVGG
HIV2D1    PQR-GDRGLATPQFSLWKRPPVTAFIEDQFVEVLLDTGADDSIVAGIELGSNYTPKIVGG
HIV2G1    SQR-GDRGLAAPQFSLWKRPPVTAYIEVQFVEVLLDTGADDSIVAGIELGSNYTPKIVGG
Smanga_S4 -QG-GDRGFAAPQFSLWRRPVTAYIEEQFVEVLLDTGADDSIVAGIELGSNYTPKIVGG
Smanga_SP -QG-GNGGFAAPQFSLWRRPVTAYIEEQFVEVLLDTGADDSIVAGIELGSNYTPKIVGG

* : * * .
```


Exercise 3 – Part 2

Tree reconstruction with RAxML

While the program is running, can you identify what parameters we are using?

-s

-n

-m

-f

-N

-x

-p

Exercise 3 – Part 2

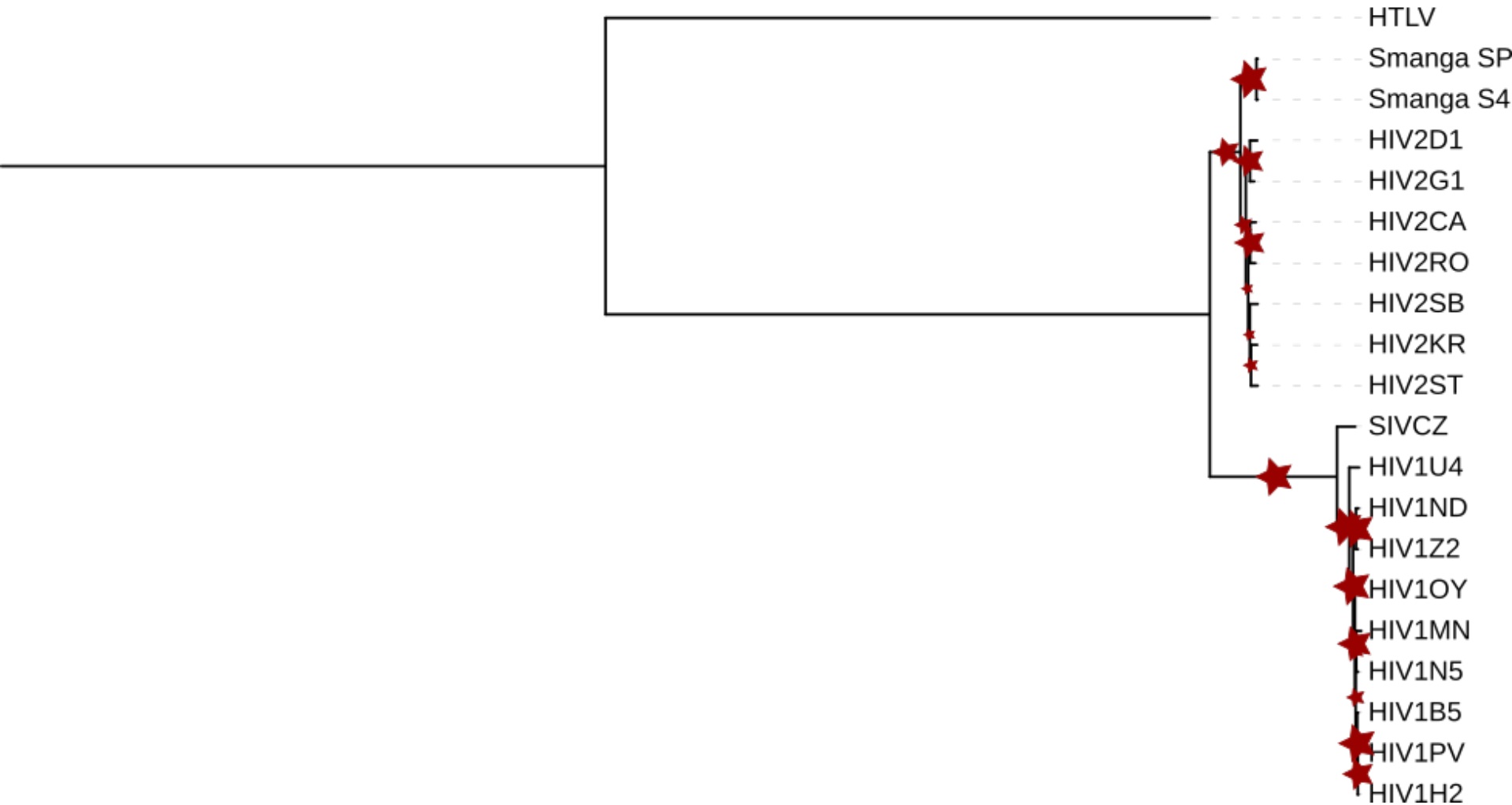
While the program is running, can you identify what parameters we are using?

- s name of the alignment data file
- n name of the output file
- m Model of Binary (Morphological), Nucleotide, Multi-State, or Amino Acid Substitution
- f select algorithm
- N number of alternative runs on distinct starting trees
- x random seed for rapid bootstrapping
- p random number seed for the parsimony inferences

Exercise 3 – Part 3



Exercise 3 – Part 4



Exercise 3 – Part 4

