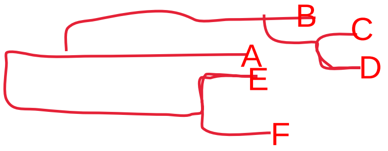
# Bio334 – 16.05.2018 – Phylogenetic reconstruction

rooted



*Afternoon session*

## Exercise 6. The Newick format

Phylogenetic trees are most commonly represented using the Newick format. In this exercise we will become familiar with its grammar rules and learn how to use it to depict phylogenies.

**Part 1:** The grammar

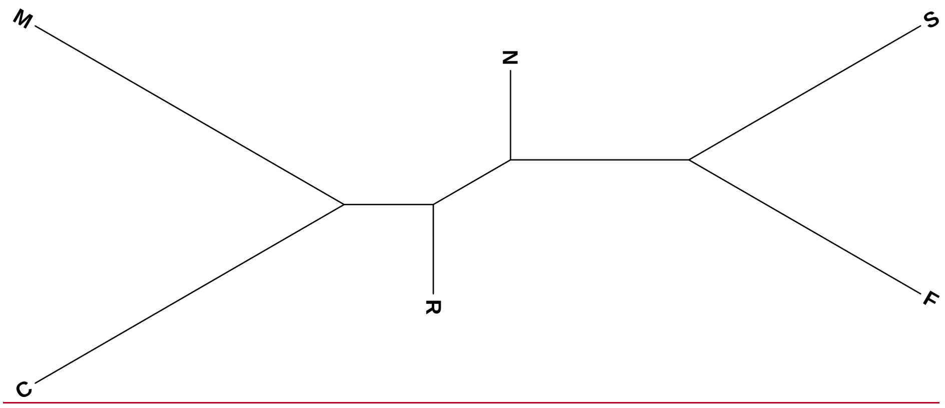
Look at a short explanation of this format on Wikipedia at http://en.wikipedia.org/wiki/Newick_format and after going through the example on the web page, try drawing the tree represented by the following Newick string:

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

Is this tree specified as rooted or unrooted? (In doubt learn more about this concept here: https://en.wikipedia.org/wiki/Phylogenetic_tree#Types). You can test your representation by uploading the example to iTOL (Tipp: you can directly write your Newick string in the "Tree text" form)

**Part 2:** Specifying branch length

The above example tree does not include any information about the length of the branches, that is, they are not scaled. As mentioned in the Wikipedia example, the Newick format allows to add branch information by adding a number representing the distance after the node's name, separated by a colon ":". To test if you understood the concept, try writing the Newick code for a tree with branch lengths proportional to the ones in the depicted tree below. Visualize your tree with iTOL to compare (select Display mode "unrooted" in the top panel).



((N:0.1,(S:0.4,F:0.4):0.1):0.1,(R:0.1,(M:0.4,C:0.4):0.1);

**Part 3:** Common errors in Newick representations

Can you identify errors in the following Newick strings? Be aware that iTOL might load them anyways

a) `(A,(E,D)),(C,(B,F));` most outer (....) forgotten: ((A,(E,D)),(C,(B,F)));

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

needs () and not []

## Exercise 7. Reconstructing phylogenies with maximum likelihood methods

**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 $\mathbf{L}_{\mathbf{max}}$. It works as following

**LOOP OVER**

**Generate tree topology ➔ Optimize branch lengths ➔ Retain if result improved**

In this session we will explore the maximum likelihood method for constructing a phylogenetic tree to investigate evolutionary relationships.

**Part 1**. Download and install RAxML

RAxML (Randomized Accelerated Maximum Likelihood) is a freely available tool for Maximum Likelihood based inference of phylogenetic trees. We will use this tool for our exercises.

The software is available from the official code repository of RAxML on the very popular code sharing website GitHub: https://github.com/stamatak/standard-RAxML. To download the latest version, click on "clone or download" and then "Download zip". This will create a folder "standard-RaxML-master" in your downloads folder or home directory. To finalize the installation, follow these steps:

a. Using the terminal locate the folder and move it to your home directory. If the downloaded file is still in ".zip" compressed format, you can unzip it using the command

```
unzip standard-RAxML-master.zip
```

b. Next, to obtain an executable from the downloaded source code, we need to compile the project. Type the following in your terminal.

```
cd standard-RAxML-master
```

(i.e. make RAxML your current directory)

```
make -f Makefile.gcc
```

(This compiles the RAxML executable)

c. This should create an executable with the name "raxmlHPC". Use the following command to acquaint yourself with the arguments available with this tool by typing

```
./raxmlHPC -help
```

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

-s sequenceFileName -n outputFileName -m substitutionModel

**Part 2.** Using RAxML on the MFS-1 dataset

Using the alignment file with the modified sequence identifiers (taxids) from the previous sessions try performing a maximum likelihood inference with the following parameters:

| | |
|---|---|
| - Input sequence: | mfs_domain_proteins.aligned.**taxids**.fa |
| - Output file extension: | mfs.auto.txt |
| - Substitution model: | PROTGAMMAAUTO |
| - Random seed: | 123 |

Differently from UPGMA and NJ, ML computations are very expensive, so even this small tree might run for a few minutes. While waiting for the results, look at the RAxML manual (included in the downloaded zip file / directory under Manual) to find out more about the substitution model we used.

**Part 3.** Visualize the RAxML results on iTOL and root the tree

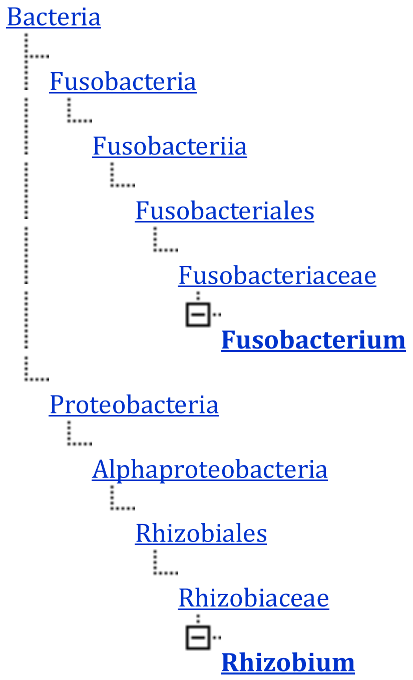
When the computations are finished you will find the results in the RAxML directory with RAxML_bestTree.mfs.auto.txt.

Upload the file to iTOL and apply the Taxonomic identifier substitution as in the morning session.

You will presumably first see a rooted tree. Note that the root has been **randomly selected**. A more realistic view is the un-rooted tree, which you can select at the top of the “Basic” tab. In order to root, we need to click on a branch and select *Editing* > *“Tree Structure”* > *“Reroot the tree here”*

Before you apply any rooting, take a moment to think where it would be best to place the root of the tree. You can help your hypothesis by looking at the Taxonomy of the species using the “Common Tree” feature of the NCBI (https://www.ncbi.nlm.nih.gov/Taxonomy/CommonTree/wwwcmt.cgi)

By inserting a few species names, you will obtain a rough idea to which domain they belong. Click on the plus to visualize the intermediate levels. E.g. by inserting “Fusobacteria” and “Rhizobium”:



- Bacteria
  - Fusobacteria
    - Fusobacteriia
      - Fusobacteriales
        - Fusobacteriaceae
          - **Fusobacterium**
  - Proteobacteria
    - Alphaproteobacteria
      - Rhizobiales
        - Rhizobiaceae
          - **Rhizobium**

After rooting the tree try to compare the resulting tree to the UPGMA and NJ trees (also root these in the same branch). What are the main differences you can see? How do the branch lengths compare in the ML version of the tree? Do the clades agree?

## Exercise 8. HIV Case Study

### Data Set Description

In this exercise you will analyze the phylogenetic relationship between HIV-related viruses from humans and monkeys: The "Pol" gene, which is present in the genome of all these viruses, encodes for polypeptides important for the viral life cycles. Here we have a data set consisting of 21 different POL-polyprotein sequences from HIV1, HIV2, chimpanzee SIV, *Sooty mangabey* SIV, and HTLV-1. The file name is `Pol21.fasta`

**Part 1.** Sequence alignment

As every phylogenetic tree building software requires a multiple sequence alignment (MSA) as an input, we will first generate the MSA using ClustalX. Save the alignment as `Pol21.aligned.fasta`.

- Load the sequences
- Do complete alignment
- Check result visually in ClustalX
- Save the alignment as fasta

**Part 2.** Tree reconstruction with RAxML

Next, for constructing the maximum likelihood tree, run the following command

```
./raxmlHPC -s Pol21.aligned.fasta -n pol21.txt -m PROTGAMMAHIVBF -f a -N 10 -x 123 -p 123
```

*NOTE: be aware that copying the command can introduce symbol artifacts which sometimes are very difficult to spot, for best results quickly re-type the command yourself*

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

-s  
-n  
-m  
-f  
-N  
-x  
-p

Which substitution model is used? Why could this parameter be particularly important for HIV sequences? More information on this topic here: https://doi.org/10.1371/journal.pone.0000503

How does the -f parameter affect the algorithm RAxML is using and why does RAxML provide different algorithms? Finally, what are seed numbers and what role do they play in stochastic algorithms? (google is your friend:-).

*NOTE: The execution of this command can take a few minutes more, feel free to take a break.*

Once the program stops running all the output files have been created in the RAxML directory. Open RAxML_info.pol21.txt to identify the number of bootstraps performed for our tree. What is bootstrapping? What makes bootstrapping necessary and what are its limitations?

**Part 3**. Tree visualization with iTOL

Load the Maximum Likelihood tree with bootstrap support measures (i.e. RAxML_bipartitions.pol21.txt) in iTOL. This time we will not apply the taxonomy identifier mapping but instead focus on visualizing the bootstrap measure that was added in this exercise:

a) In the “Basic” tab, see if you can invert the label position (from right to left) as well as increase the branch thickness to make the tree more readable.
b) Now to the bootstrap values. In the “Advanced” tab, choose to “Display” the Bootstraps/Metadata section. Explore the options of the new sub-menu. Can color the branches according to bootstrap? What do red branches symbolize in this representation?

**Part 4**. Rooting the tree with an outgroup

Now we will learn to make a rooted tree with the help of an outgroup. We know from other sources that the lineage leading to HTLV branched off before any of the remaining viruses diverged from each other. Therefore, the root of the tree connecting the organisms being investigated must be located between the HTLV sequence (the "outgroup") and the rest (the "ingroup"). This way of finding a root is called "outgroup rooting".

a) Use the re-root option and root it after selecting the HTLV branch.
b) Can you describe the evolutionary relationship depicted by the tree? What are the sister clades of HIV1? *Sooty mangabey* is most closely related to which HIV type?
c) What do high bootstrap values mean?
d) How can you asses the robustness of the tree using the bootstrap values? What likely divisions do the high bootstrap values support?