

# Lab 6 – Phylogenetic Analysis

To get the lab approved, send your answers to: [david.sundell@plantphys.umu.se](mailto:david.sundell@plantphys.umu.se)

This practical uses two programs, a simple alignment and neighbor joining tree building program, ClustalX, and FigTree, a very useful tree viewing program. Clustal is available from a number of different servers, e.g., <http://www.clustal.org/>. FigTree is available from its own website, <http://tree.bio.ed.ac.uk/software/figtree/>.

## Neighborjoining distances

This exercise concerns a famous criminal case in Florida in 1990 where a dentist with HIV was accused of infecting some of his patients during routine dental procedures (back in the bad old days before everyone used surgical gloves). Part of the evidence at the trial was a phylogeny of HIV, including sequences from the dentist, infected patients and controls. You will re-analyse these data and decide if you think that the dentist was guilty or not.

### Part 1: Align sequences

**1.Retrieving the sequences:** First get the sequences, through NCBI (**Nucleotide database**) or EMBL (using SRS). You can find the following sequences using three search terms: “**Ou**” and “**Ciesielski**” (the authors) and “**V3**” (the fragment of the HIV envelope protein gene used in the study). Accession names are annotated as follows – HIVFL stands for HIV FLorida, P for patient (multiple sequences from same patient are named PA, PB, etc.), D for dentist and Q for controls.

Select the following:

1. All isolates from the dentist:

HIVFLD1, HIVFLD2, HIVFLD4, HIVFLD5, HIVFLD7, HIVFLD8

2. Several isolates from each of the infected patients

HIVFLPA5, HIVFLPA6, HIVFLPB3, HIVFLPB7, HIVFLPC12, HIVFLPC14, HIVFLPD1, HIVFLPD9, HIVFLPE6, HIVFLPED HIVFLPF5, HIVFLPFD,HIVFLPG3, HIVFLPG4, HIVFLPH4D, HIVFLPH7D

3. A set of control sequences (outgroup, maybe)

HIVFLQ18, HIVFLQ31, HIVFLQ34, HIVFLQ38, HIVFLQ44, HIVFLQ66,HIVFLQ69, HIVFLQ710, HIVFLQ77

Tick the boxes for all the accession numbers listed above and save them together in a single file. Reformat the sequences by changing the Display from Summary to FASTA and then sending the data to a text file. Finally, save the output as a file, using the file menu in your browser.

## ... but wait a minute ... **We can do this automatically with BioPerl!**

Run the Perl program `get_sequences.pl` with the file `hiv_ids.txt` that contains the accession names. The sequences will be put into `hiv.fasta`. See how nice it is to know how to program?

**2. Sequence names.** It will make your life a lot easier if you simplify the sequence names. Open the output file in a text editor. I suggest just retaining the sequence name. So for example `>HIVFLD1` instead of Human immunodeficiency virus type 1, viral sample FLD1, V3 region. Remember to keep the `>` which is essential for fasta format.

## ... but wait a minute ... **We can do this automatically with Perl!**

Run the Perl program `change_names.pl`. It simplifies the names in `hiv.fasta` and make a new file called `hiv2.fasta`. See ... programming is king!

**3. Align the sequences.** Before building the tree, you must first align the sequences. Launch ClustalX and import your sequences (Load Sequences in the File menu). Next align the sequences by selecting Do Complete Alignment in the Alignment menu. How does it look? Do you see any potential problems? In particular, consider what will happen when you later will use the Clustal command to delete positions with gaps. Maybe it's a good idea to remove HIVFLPED not to lose too much information (Edit -> Cut Sequences).

### **Part 2. Building the tree.**

**1. Selecting parameters.** Before constructing the tree you must decide which parameters to use in the analysis. First, since there are gaps in the alignment and incomplete sequences (missing ends) you will want to delete these regions from the analysis. To do this check the Exclude Positions with Gaps option in the Tree menu. Secondly, look at the alignment and decide if you need to correct for multiple substitutions, i.e., are the sequences different enough from each other so that there are hidden changes?

**2. Constructing the tree.** Now calculate the tree. First you must reformat the program's output to make it compatible with just about any treeviewing program (e.g.,

FigTree). From the Trees menu, select Output Format Options and change Bootstrap labels on from Branch to Node. Forgetting to do this is the most common mistake people make with Clustal. If you open your tree in a treeviewing program and there are no bootstrap values, this is probably the reason. Still in the Trees menu, select Bootstrap NJ Tree. This will open a new window in which you can select the number of bootstrap trials. You should set this number to at least 1000 – with bootstrapping, more is always better. NB: at this point the program also offers you a default file name for your output files. You will probably want to rename these (the simplest thing to do is add a number to the end of the name). If you run multiple analyses with this same alignment (and you often will), subsequent output will not overwrite each other. Now click OK, and the analysis will begin automatically.

**3. An important point.** While the analysis is running, you should consider the following. The analysis that you are running actually consists of two separate analyses. These are 1) a neighborjoining (NJ) tree analysis (based on the entire alignment), and 2) a bootstrap analysis (based on multiple random subsets of your alignment). The final output that you will get is a combination of these two. That is, the Clustal “Bootstrap NJ tree” output is the NJ tree (derived in step 1) combined with the bootstrap numbers (calculated in step 2). It is important to understand the difference between an NJ tree and a bootstrap analysis.

### Part 3. Viewing the results

**1. The tree.** To view your results, launch FigTree. Go to the File menu, select Open, and search for a file with the extension .phb. That’s the output from your ClustalX analysis. When you click on the file you’ll get a pop-up window asking you to name the values generated by your analysis. These are bootstrap values, so type bootstrap (or just boot) and enter. You should see your tree now. In order to see it better, you can increase the width of the lines in the Appearance menu (click on the arrow). You can also increase the font size (and change font) of the names in the Tip labels menu. Finally, root the tree with one of the control (outgroup) sequences. I suggest you use HIVFLQ77, which is the sequence they used in the original paper. Click on the branch separating this sequence from the rest of the tree and then select Reroot icon at the top of the window.

**2. Interpreting your results.** So what do you think? Do you think he was guilty? How confident of your conclusion are you? What do you need to be more confident? Now display the bootstrap values using the Node Labels menu. Tick the box, open the menu and change the display from Node ages to bootstrap (or boot, which ever you used). Does this make you feel more or less confident?

Questions:

1 Was the dentist guilty?

2 On all counts?

3 How confident are you?

4 How do you think they selected the outgroup for the study (i.e., what were the controls for this experiment)?

5 What if HIVFPED was an important sequence? What could you do to keep it in the alignment without it causing you to lose so much useful information from the other sequences?

4. Annotating your tree. FigTree has lots of options for annotating trees. For instance, you can color all the dentist sequences, or all the patients, or use different colors for each. You can color either names or lines or both. You can collapse some of the nodes, e.g., the outgroup, so you can focus more on the ingroup (maybe not so useful for this tree, but useful to know how to do).

5. Tree figure. You can create a figure from this tree in Word. Output the tree as a graphic, using .png or .svg. Now you can import it into Word and add a legend or annotate it further.

Read more in the paper: Molecular Epidemiology of HIV Transmission in a Dental Practice, Science 256: 1165–1171, 1992.