Sequence bioinformatics, computer exercise XII
University of Potsdam, WS 2023/2024, Stefanie Hartmann

Give it all you've got! And since the mere presence of smartphones reduces available cognitive capacity (`https://www.journals.uchicago.edu/doi/full/10.1086/691462`), I suggest you put your cell phone out of sight and into your bag for the duration of this computer lab.

Inferring phylogenies

Objective

The goal of this and next week's exercise is to learn how to compute, view, read, and interpret a phylogenetic tree. Today you'll compute a neighbor-joining tree and corresponding bootstrap values. Next week you'll compute a maximum likelihood tree for the same sequences, and you'll spend more time analyzing and interpreting both phylogenies then.

1 Locally installed software you'll need

To properly view a phylogeny, you'll need a tree viewer. I recommend the program FigTree, available for all operating systems at `https://github.com/rambaut/figtree/releases`.

If you wish to view the alignment I've provided, you'll also need the alignment viewer (e.g., JalView) you already used during the last two labs.

2 Logging into the cluster and preparing for the analyses

☞ Log into the university's computing cluster as described before and change to your directory for this class on the /work partition. Create a new directory for today's exercise (e.g., lab12) and change into it.

☞ You'll use the Phylip software today. Load the corresponding module with the command `module load bio/PHYLIP/3.697-GCC-9.3.0`

☞ Start an interactive session with the command `srun --cpus-per-task=1 --mem=5G --pty bash`

3 The data

For today's computer lab you'll use the same set of sequences that you worked with two weeks ago (MSA lab). I'm providing an alignment of these sequences that is profileHMM-based and that is in phylip format. Since the phylip format doesn't allow long sequence identifiers, I have truncated them for the purpose of this exercise. Note: If you decide to compute a second phylogeny from a different alignment (e.g., one of your own alignments from last week or from the week before), you need to convert the fasta or stockholm file into phylip format. This can be done using the seqret conversion tool that was described last week.

☞ The file "tocs75.hmm.phy" is available in the shared folder, copy it into a directory for today's exercise and briefly look at it with `less`. Briefly describe the phylip format in your own words, afterwards look it up, e.g., at `http://rosalind.info/glossary/phylip-format/`.

☞ Use linux system utilities to find out how many sequences the file contains, from how many species, and how many sequences per species. The codes for species are as described in lab09.

☞ If you like, copy the alignment to the local computer and view it with an alignment viewer for closer inspection.

4 The Phylip package

Phylip is a freely available package of programs for doing various kinds of phylogenetic analyses. The Phylip homepage at `http://evolution.genetics.washington.edu/phylip.html` has information about each of the programs. Included are, for example, programs for inferring phylogenies of from DNA and protein data using both distance- and character-based methods. It also contains programs for computing consensus trees (e.g., from bootstrap data sets) and tree manipulation (e.g., re-rooting a tree).

4.1 Input and output

- by default, most of the Phylip programs take a file named 'infile' as input. If there is no file called 'infile', you will be asked to specify an alternative file to be used as the input file.
- input sequence alignments need to be in Phylip format, otherwise the programs do not accept them and will exit with an error message.
- program output is generally written into files named 'outfile' and/or 'outtree'. If such a file already exists, the program asks whether the existing file should be overwritten, or whether a file with another name should be created.
- output trees are written in newick format.

Note: because you will run several Phylip programs one after the other, you need to be careful to rename the intermediate files. The 'outfile' from one program will become the 'infile' for the next program. It is very easy to overwrite existing outfiles and/or get confused as to which program generated a given file called 'outfile'. It is therefore a good idea to rename the outfiles right away and give them more meaningful names.

4.2 Running Phylip programs (general)

The Phylip programs can be started from the terminal by typing the name of the phylip program you want to run. For example, the Phylip program 'protdist', which computes a distance matrix from protein sequences, can be started using the command `protdist`. All Phylip programs run interactively in the terminal: a text-based menu, like the one shown below for protdist, is available for each program. This can be used to set options and to start the program.

```
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

Are these settings correct? (type Y or the letter for one to change)
```

To change any parameter, you type the letter under "Settings". For example, to change the substitution model for the estimation of distances from the default JTT (Jones-Taylor-Thornton) to another one, you would hit 'P' on your keyboard repeatedly, until the model of our choice is listed in the right-hand column. Once all settings are what you want them to be, type 'Y' to start the analysis.

5 Neighbor Joining analysis

5.1 Computing the distance matrix

1. start `protdist` by typing its name. The program will hijack your terminal program, and only protdist commands are now allowed/possible.
2. since there is no file called 'infile' in this directory, you will be asked to enter the name of the file. Note that tab-completion doesn't work within these programs; you'll have to type out the name of the files
3. the default substitution model for computing the distance matrix is the Jones-Taylor-Thornton (JTT) model. This is very similar to Dayhoff's PAM model but based on a larger data set. To accept all defaults, type 'Y' and hit Return to start the program.
4. the program computes the pairwise distances, tells you that the file 'outfile' was generated, and exits.

First rename the outfile and give it a more meaningful name, then look at the distance file using `less`. There is no standard way to name the a distance file. Use the extension `.dist` or `.txt`, for example.

☞ Briefly describe the format of the distance file. Note that Phylip provides output for small terminal windows (regardless of how wide you set your's), and so each matrix line is split into multiple lines, depending on the number of sequences.

☞ Are the distances above and below the diagonal identical? yes

0.3-2.7

☞ Just briefly looking at the file, what is the approximate range of distance values? Are the distances generally in agreement with what you would eyeball from the multiple sequence alignment? i think so..?

5.2 Computing the neighbor joining tree

To compute the neighbor joining tree, use the Phylip program `neighbor` in a similar fashion:

1. start `neighbor`
2. since there is no file called 'infile' in this directory, you will be asked to enter the name of the file. Use the distance file you just generated.
3. different settings are available. Briefly look at them, but just accept the default settings and type 'y', followed by a Return to begin the analysis.
4. the program computes the phylogeny, lets you know that the files 'outfile' and 'outtree' were generated, then it exits.

Again, before doing anything with these files, rename them so you can later remember what they are! Although `.tre` is one of the commonly used extensions for tree files, there is no standard way to name the other output file. Use the extension `.nj`, for example.

☞ Take a closer look at the .nj file. It shows a graphical representation of the tree, tells you that the tree is unrooted, and it lists the branch lengths of all branches of the tree. Internal nodes have been arbitrarily named for the list of branch lengths.

☞ Which is the shortest branch on the tree? Which is the longest branch on the tree? 3/ath_174 and gsu_005

☞ Take a look at the .tre file. This file contains the tree in newick format. It contains information about the tree topology and branch lengths. This is a little difficult to read, and so in a moment you'll use the FigTree program on your local computer to view the tree.

음수값을 가진 branch length가 있는데, 이것은 직접적으로
neighbor joining methode가 problematic이고 완전히 믿어서는
안된다는것을 의미한다. 이 메소드는 결국 수학적으로
꼐산하기 때문에, 수학적으로 말이 될 수 있어도,
생물학적으로는 말이 안된다는것을 의미한다.

☞ How does the newick format represent branch lengths?

5.3 Viewing the tree with Figtree

☞ Use `scp` to copy the tree file to your local machine, then open it within the the FigTree program.

☞ The default display is a rectangular tree layout. You can increase the font size for better readability in the menu on the left, under "Tip Labels": view the options by clicking on the triangle on the left, then click to increase the font size.

☞ The cyanobacterial sequences (pro_) can be used as outgroups. Find the branch that separates these two from all the other sequences, then click on the "Reroot" button in the top menu.

☞ Explore and adjust other display settings if you like.

6 Computing bootstrap support values

How robust is the clustering of the sequences? To be able to assess confidence in the clades of the NJ-tree, you will compute 100 bootstrap replicates and repeat the computation of distances and a tree for each one of them. This will be very similar to what you just did, but a step at the beginning will be added to generate the bootstrap data sets (pseudosamples), and another extra step at the end will analyze the bootstrap trees.

Please note that computing bootstrap values is not a distinct or stand-alone means to infer a phylogeny: it only evaluates confidence in the clades you just computed and should be viewed and interpreted in the context of that analysis.

6.1 Generating 100 bootstrap data sets

The phylip program `seqboot` takes a multiple sequence alignment as input and produces one or more new alignments with replacement from the columns in the original alignment. It works just like the other phylip programs do.

1. start the program `seqboot`
2. use the trimmed alignments as input, in phylip format
3. use all default parameters: "Y" (the default generates 100 replicates)
4. enter a random odd number; this number seed is used to start the random number generator for the resampling. If the random number seed is not odd, the program will request it again
5. rename the new file 'outfile' to a more appropriate name. View this file to confirm that it contains many (100, to be exact) sequence alignments in phylip format, all one after the other

yes

6.2 Computing 100 distance matrices

Next compute a distance matrix for each of the 100 new alignments. You can do this with the `protdist` program you used before.

1. start the program `protdist`
2. use output from seqboot as input
3. use the option 'M' to specify that you want to analyze multiple data sets. Tell the program that you have data sets ('D'), and that you have 100 of them
4. accept all settings and start the analysis. This will take 100 times as long as the original run, but it will still go fairly quickly.
5. rename the outfile and take a look at it using `less`

6.3 Computing 100 trees

Run `neighbor` with the distance file you just generated:

1. start the program `neighbor`

2. use the output from protdist as input
3. use the option 'M' to specify that you want to analyze multiple data sets. Tell the program that you have 100 of them.
4. accept all settings.
5. provide a random number for the seed and start the analysis. This will take 100 times as long as the original run, but it will go very quickly.
6. rename the outfile to something more appropriate and take a look at it using `less`. It contains trees and branch length information for all 100 bootstrap data sets.
7. rename the outtree to something more appropriate and take a look at it using `less`. It contains 100 bootstrap trees in newick format. It's not easy to read; a new tree starts after the line that ends with a semicolon.

6.4 Compute a consensus tree from the 100 bootstrap trees

The final step is to use the `consense` program to compute a consensus tree from the 100 bootstrap trees.

1. start the program `consense`
2. use file with the 100 trees in newick format as input
3. the default parameters are all ok, and so you can just accept them to start the analysis
4. rename the outfile to something more appropriate and take a look at it using `less`. At the bottom of this file (and that's all that's important for now), you see a graphical representation of the consensus tree. The weights on the branches indicate the number of bootstrap trees that had a branch resulting in the same partition of sequences on either side ("bipartitions"). More information about the output is available at `https://evolution.genetics.washington.edu/phylip/doc/consense.html`
5. rename the outtree to something more appropriate and take a look at it using `less`. It contains the bootstrap consensus tree in newick format.
   ☞ Can you figure out how the bootstrap values are represented in the newick format? Compare this to how the branch lengths were stored in NJ tree you computed earlier today.

6.5 Viewing the consensus tree with Figtree

☞ FigTree can display the bootstrap support values you just computed. Use `scp` to copy the consensus tree to your local computer and open the file with FigTree. In the menu on the left, click on the box next to "Node Labels". Open the "Node Labels" option and select Display: Branch lengths (raw).

☞ Move/resize the FigTree windows so that you see the original NJ tree and the consensus tree side by side on your screen. Compare the two. Which clades are well supported, which are poorly supported? Are all bipartitions of the original NJ tree also seen in the consensus tree?

> Note: For easier comparison, you can also re-root the bootstrap consensus tree with the prokaryotic sequences, although this will lead to a different (displayed!) bootstrap value on this branch. Also, select in the "Tree" menu the option "Order nodes: increasing" for easier comparison.

Although this is a little cumbersome, you'd have to transfer the bootstrap values from the consensus tree to the original NJ tree, for example using graphics software. Next week you'll use a different program to compute a tree, and the bootstrap values will be placed onto the best tree automatically by the program.

As mentioned at the beginning, the in-depth analysis and interpretation of the inferred phylogeny will be done next week. However, please take a look at today's tree and its support values and start to think about the evolution of these genes/sequences.

7 Exiting the terminal

Log out of the interactive session with `exit`.

Then log out of the current session to the cluster, type `exit` again.

Finally, to cleanly close the terminal window, type `exit` yet again.

Do not close a terminal window with the mouse, especially not if an ssh session is still active.

➜ Don’t forget to take your flash drive with you when you leave!