Week 5: Aligning with MUSCLE and Making Trees with RAxML

Rika Anderson

Abstract

Citation: Rika Anderson Week 5: Aligning with MUSCLE and Making Trees with RAxML. **protocols.io** dx.doi.org/10.17504/protocols.io.g2sbyee
Published: 04 May 2017

Protocol

Intro for this week

Step 1.

This week in lab we're going to learn how to make phylogenetic trees. We'll start by creating a multiple sequence alignment with MUSCLE, and then we will make bootstrapped maximum likelihood phylogenetic trees with RAxML. We'll use the Newick files generated by RAxML to visualize trees in an online tree visualization tool called the Interactive Tree of Life (iToL). We'll do this using toy datasets, and then try out some trees on your project datsets.

First, log on to liverpool.

A. Aligning sequences

Step 2.

First we have to make a multiple sequence alignment with the sequences we wish to make into a tree. This could include any gene of interest. We're going to start by aligning a toy dataset made of genes that are part of photosystem II in photosynthesis. This file contains many sequences of the same photosystem II protein from different species.

Make a new directory within your toy dataset directory for making alignments and trees.

cmd COMMAND

```
mkdir toy_dataset_directory/alignments_and_trees
cd toy_dataset_directory/alignments_and_trees
```

A. Aligning sequences

Step 3.

Copy the toy dataset from the data directory to your toy dataset directory.

cmd COMMAND

```
cp /usr/local/data/toy_dataset_PSII_protein.faa .
```

A. Aligning sequences

Step 4.

Now, we make a multiple sequence alignment using muscle. It’s remarkably easy and fast.

What this means:

1. muscle is the name of the program
2. –in defines the name of your input file, which can be either DNA or protein
3. –out defines the name of your output file. I like to give them an easy-to-recognize name with the extension “.afa”, which stands for “aligned fasta file.”

cmd COMMAND

```
muscle -in toy_dataset_PSII_protein.faa -out toy_dataset_PSII_protein_aligned.afa
```

A. Aligning sequences

Step 5.

Let’s take a look at your alignment. I like to use the program Seaview to do this, and Seaview is on your local computer. So you have to copy your file to your local computer. You could use FileZilla to use this. But if you want to learn a handy Unix trick, you could use scp (secure copy). It lets you copy files to and from your local computer on the command line. You have to know the path to the file you want to copy on liverpool, and the path to where you want to put it on your local computer. It's a lot like cp, except it allows you to copy things between computers.

Open up a new Terminal window, but DON'T log in to liverpool on that one. Then type the command below. It will prompt you for your Carleton password. For this particular case, it will copy it to your Desktop, but you can specify whatever destination path you want.

cmd COMMAND

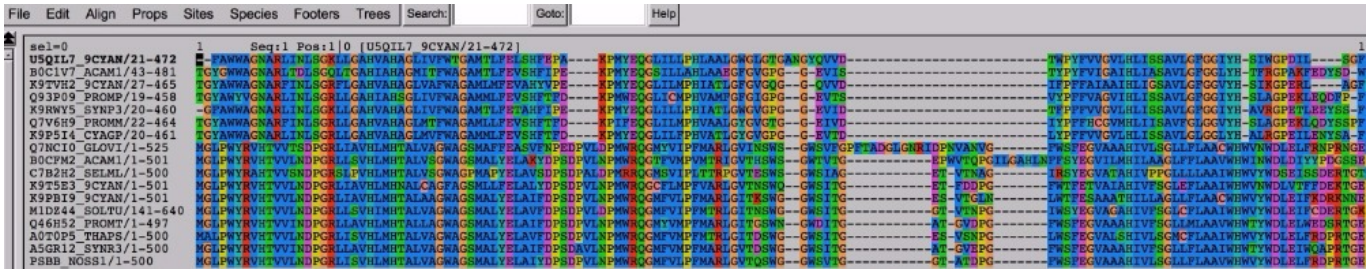
```
scp [username]@liverpool.its.carleton.edu:/Accounts/randerson/toy_dataset_directory/alignments_and_trees/toy_dataset_PSII_protein_aligned.afa ~/Desktop

scp ~/Desktop/[some_file.txt] [username@liverpool.its.carleton.edu:/Accounts/randerson/toy_dataset_directory/
```

First command: Securely copy a file from liverpool to your local computer. Second command: an example of how you might copy a theoretical file from your local computer to the toy directory on liverpool.

A. Aligning sequences

Step 6.

Open the application called “Seaview” and drag your file over to the alignment window. You should see something like this.



Seaview shows the names of the sequences to the left. The letters to the right are the amino acids in your sequence, color-coded to make the alignment easier to see. You can easily see that some regions of the sequence are more highly conserved than others, and that some species appear to have an insertion or deletion in specific regions of the sequence.

Pause here for a moment and draw a sketch of what you predict the phylogenetic tree will look like based on this alignment. I don't expect you to use the distance method to calculate exact distances, but draw a schematic of a tree in which you predict which sequences will fall together into the same clades. We will reconvene as a class to draw those trees on the board, and then in the next few steps you will make a tree computationally to see how they compare.

B. Creating phylogenetic trees with RAxML

Step 7.

Now we're going to turn this alignment into a phylogenetic tree. We're going to use a software package called RAxML, which is a commonly used tree-building software package that uses the maximum likelihood method to build trees.

First, we have to convert the muscle output, which is an aligned fasta file, into a format that the tree-building software (RAxML) can handle. (A lot of the work of a bioinformatician is converting files from one format into the other.) I wrote a small Python script to convert your aligned fasta (.afa) file into Phylip file (.phy), which is what RAxML requires for input. Invoke it with the name of the script, followed by the aligned fasta file that you'd like to convert, like this:

cmd COMMAND

```
convert_afa_to_phy.py toy_dataset_PSII_protein_aligned.afa
```

B. Creating phylogenetic trees with RAxML

Step 8.

The script should have outputted a file that ends in .phy. Take a look at it to see what the format looks like.

The top of a Phylip file indicates the number of sequences (in this case, 17) and the number of base

pairs in the aligned sequences (571). It's followed by the first 54 bases of each sequence, broken up by a space every 10 letters. Every new line shows a new sequence (from a different species or organism). The next 54 letters of each sequence starts after the first set, and continues like this for the rest of the file. (Yes, it's weird, but a lot of tree-building programs use this format.)

cmd COMMAND

```
less toy_dataset_PSII_protein_aligned.phy
```

B. Creating phylogenetic trees with RAxML

Step 9.

Now let's make a tree. Type this:

What this means:

-raxmlHPC-SSE3 is the name of the software package. This one is configured for the processors that are specific to this server.

--f a allows for rapid bootstrapping. This means RAxML will do a maximum likelihood search based on your protein sequences, and then bootstrap it as many times as you wish.

-# 20 tells the program to bootstrap 20 times.

-m PROTGAMMAAUTO tells the program that these are protein sequences, and tells the program how to model protein evolution. To get into this is beyond the scope of this class, but fortunately RAxML is able to automatically choose the best one for us based on the number of sequences and the type of data we have.

-p and –x provide seed numbers so that the program can generate random numbers for the bootstrapping process.

-s gives the name of your input Phylip file

-n gives the name of your output Newick file, which will be made into a tree.

cmd COMMAND

```
raxmlHPC-SSE3 -f a -# 20 -m PROTGAMMAAUTO -p 12345 -x 12345 -
s toy_dataset_PSII_protein_aligned.phy -n toy_dataset_PSII_protein.tree
```

B. Creating phylogenetic trees with RAxML

Step 10.

You've made a tree! Let's look at the raw RAxML output. You should have some files called:

RAxML_bestTree.toy_dataset_PSII_protein.tree

RAxML_bipartitionsBranchLabels.toy_dataset_PSII_protein.tree

RAxML_bipartitions.toy_dataset_PSII_protein.tree

RAxML_bootstrap.toy_dataset_PSII_protein.tree

RAxML_info.toy_dataset_PSII_protein.tree

B. Creating phylogenetic trees with RAxML

Step 11.

The one you want is called:

RAxML_bipartitions.toy_dataset_PSII_protein.tree

Because it gives you bootstrap values at each of the nodes of the tree. Take a look at that file using less.

This is a Newick file, and it's a common format for phylogenetic trees.

B. Creating phylogenetic trees with RAxML

Step 12.

Let's look at the Newick file in its visual form. Copy your Newick file (toy_dataset_PSII_protein.tree) to your local computer using FileZilla or scp.

B. Creating phylogenetic trees with RAxML

Step 13.

Open up a web browser on your local computer and navigate to http://itol.embl.de/ and create an account for yourself.

B. Creating phylogenetic trees with RAxML

Step 14.

Click on "Upload tree files" and find your tree file and upload it.

Click on your tree. You should see it open in your window. You can play around with the settings on the right—you can make it circular or normal, you can display the bootstrap values as symbols or as text, and if you click on the leaves (the tips) or the nodes (the interior bifurcations) of the tree, you can color code them. Play around with it, and then click on the "Export" tab, choose the PDF format, and click "Export." It should pop up in a new tab. **Download it and include it in your lab writeup this week as "Figure 1."**

C. Asking Biological Questions with Trees

Step 15.

Now that you know how to make an alignment, create a phylogenetic tree, and visualize it, we're going to ask a biological question that can be answered with trees:

To which organisms are photosynthesis genes in surface waters most closely related?

We would probably expect to see more photosynthesis genes related to photosynthetic bacteria, and perhaps seaweed, in the surface oceans, but not genes related to terrestrial plants. We can make a phylogenetic tree to determine whether that's true, and which specific bacteria they're most closely related to.

C. Asking Biological Questions with Trees

Step 16.

First, we need to identify a target protein within the photosynthetic apparatus. For this, let's go to the KEGG Pathways website, which has a description of major metabolic pathways and the proteins that comprise them.

LINK:
http://www.genome.jp/kegg/pathway.html

C. Asking Biological Questions with Trees

Step 17.

Click on 'Energy' under 'Metabolism' and then click on 'Photosynthesis.' You should see a depiction of the photosynthetic proteins. Let's choose a protein within the photosynthetic apparatus that might be of interest. For our sakes, let's choose psbA: it's part of the photosystem II p680 reaction center D protein. Click on the box labeled 'PsbA' beneath the figure. Now you'll see lots of information about that particular gene. You can use the KEGG website like this to figure out which proteins or genes might be of interest for your projects later on.

C. Asking Biological Questions with Trees

Step 18.

However, it will be more useful for us to have a nice set of seed sequences to align against-- so let's go back to Pfam. Google 'psbA pfam' and click on the first link (or click on the link below).

LINK:
http://pfam.xfam.org/family/PF00124

C. Asking Biological Questions with Trees

Step 19.

Click on '708' sequences near the top click on 'FASTA' from the dropdown menu, click 'No gaps (unaligned)' from the drop-down menu, and save the resulting file on liverpool in your project dataset directory. It would probably be best to create a separate folder for this. Use FileZilla or scp to transfer it over to that new directory, or you could copy the file and create a new file on liverpool using nano and copy it there (many options!).

cmd COMMAND
```
mkdir project_directory/tree_exercise_1
cd project_directory/tree_exercise_1
```

C. Asking Biological Questions with Trees

Step 20.

First we have to find matches to that gene in our dataset. So, we have to BLAST this dataset against the dataset of interest. We're going to use an assembled Tara Oceans dataset from the North Atlantic surface oceans off the East Coast. Copy it over to your current directory.

cmd COMMAND
```
cp /usr/local/data/ERR598983_ORFs.faa .
```

C. Asking Biological Questions with Trees

Step 21.

Make this dataset into a BLAST database.

cmd COMMAND
```
makeblastdb -in ERR598983_ORFs.faa -dbtype prot
```

C. Asking Biological Questions with Trees

Step 22.

BLAST it.

cmd COMMAND
```
blastp -query PF00124_seed.txt -db ERR598983_ORFs.faa -outfmt 6 -evalue 1e-05 -
out PF00124_vs_ERR598983_ORFs.blastp
```

C. Asking Biological Questions with Trees

Step 23.

Take a look at the results. You'll notice lots of hits. However, you'll see there was a hit to a single ORF: contig-100_83_1.

cmd COMMAND
```
less PF00124_vs_ERR598983_ORFs.blastp
```

C. Asking Biological Questions with Trees

Step 24.

Let's extract it. Open up the original file with the open reading frames, and copy the sequence for contig-100_83_1.

cmd COMMAND
```
less ERR598983_ORFs.faa
```

ANNOTATIONS

Simon Orlovsky 30 Jan 2017

You can search for a specific contig by typing

/contig-100_83_1

C. Asking Biological Questions with Trees

Step 25.

Now let's add it to the original Pfam file and make a tree from all of those sequences. First, make a copy of the original Pfam file. Then use nano to open it up and paste the ORF sequence to the bottom.

cmd COMMAND

```
cp PF00124_seed.txt PF00124_seed_plus_new_ORF.fasta
nano PF00124_seed_plus_new_ORF.fasta
```

ANNOTATIONS

Dustin Michels 28 Oct 2017

You could also do this with Vim-- an alternative terminal-based text editor to Nano, which comes with some slick commands for navigating a document.

* Type '`vi PF00124_seed_plus_new_ORF.fasta`' to open the file

* Type 'G' to jump to the last line

* Type '$' to jump to the end of the line

* Type 'a' to start appending text after your cursor

* Make a new line (with [enter]) and paste (with 'cmd + v')

* To save and exit, press '[esc] `: wq [enter]`' (one at a time).

- Escape takes you out of editing mode, colon opens up the vim command prompt, and 'wq' is the command to save and exit.

C. Asking Biological Questions with Trees

Step 26.

Make a multiple sequence alignment with muscle.

cmd COMMAND

```
muscle -in PF00124_seed_plus_new_ORF.fasta -out PF00124_seed_plus_new_ORF_aligned.afa
```

C. Asking Biological Questions with Trees

Step 27.

Convert your aligned FASTA file to a Phylip file.

cmd COMMAND

```
convert_afa_to_phy.py PF00124_seed_plus_new_ORF_aligned.afa
```

C. Asking Biological Questions with Trees

Step 28.

Make your tree. (This might take about 15-20 minutes, so you might get started on the last question for the day while this runs.)

cmd COMMAND

```
raxmlHPC-SSE3 -f a -# 20 -m PROTGAMMAAUTO -p 12345 -x 12345 -
s PF00124_seed_plus_new_ORF_aligned.phy -n PF00124_seed_plus_new_ORF.tree
```

C. Asking Biological Questions with Trees

Step 29.

Visualize your tree with ITOL.

Questions to answer on the document you'll submit to Moodle:

Include an image of your tree and call it 'Figure 2.'

1. Which organisms have the three most closely related psbA proteins to your ORF? (Hint: you can use UniProt, as we learned last week, to figure out what organisms these proteins come from.)

2. Based on these reults, what might you infer about the photosynthetic organisms in your sample? Are they eukaryotes, archaea, bacteria? Do you think you can make broad conclusions about the whole community of photosynthetic organisms in your sample? Why or why not?

D. Asking Your Own Biological Questions with Trees

Step 30.

Come up with a question that you can answer about your own project dataset using trees. If you BLASTED a protein from last week and found some interesting hits and want to learn more about it, this could be your opportunity. Or you could ask an entirely different question.

On the document you'll submit to Moodle, describe:

- What question did you ask?
- How did you go about answering it? (Write this like you would a Materials and Methods section: include which databases you searched, which software packages you used, and which flags you used in your commands, if any. There is no need to write out the entire command you typed.)
- What were your results? Describe them. (Include a tree if appropriate.)
- What does this tell you about your project dataset? (This is an open-ended question and depends on what question you asked. Think of this as a mini Discussion section: I'm looking for evidence that you thought about your results and how they connect more broadly to some ecological or evolutionary pattern in your dataset.)

Compile Figure 1, Figure 2, the two questions in the previous step, and this week's mini-research question together and submit on Moodle by lab time next week.