**Lopatkin / Snow : PAML usage through python for nucleotide sequences**

Compiled by Emily Huntsman, Barnard ‘21

May 2021

**PART 1: file generation for use with CodeML**

Inputs: NT sequences with species names

Outputs: formatted alignment .phylip file, newick tree .trees file for use with codeML

1. Generate alignments in MUSCLE (https://www.ebi.ac.uk/Tools/msa/muscle/) specifying nucleotides as the input and sequential PHYLIP format for the output
   1. Make sure the file has the extension .fa not .txt so MUSCLE can read it
   2. Sample input file IRE1\_NT.fa
      1. Carrot symbol followed by species name (lowercase letters, uppercase letters, . , \* all ok), no two species can have the same name but can vary by one character based on an internal naming convention, the species names are those that will be used throughout the process so they must be kept constant between files
      2. On the next line, the nucleotide sequence (all NT must be on the same line of the text file, check this by opening it in a text editor)
      3. Repeat for as many species as necessary, always starting the new species on its own line
   3. Press “Download Alignment File” and copy and paste into a new file with the extension .phylip
   4. Sample output from running MUSCLE on IRE1\_NT.fa is in file MUSCLE\_output.phylip
      1. The formatting of this output file will likely have to be tweaked, an example of appropriate formatting of that output can be seen in formatted\_alignment.phylip
         1. The most notable difference here is that if you open both files in a text editor, you can see that MUSCLE separates the alignment into multiple lines which is not acceptable for the file that will be inputted into CodeML
         2. In formatted\_alignment you can see that those extra new line characters have been omitted and each set of 10 characters is separated by a single space
2. Construct a phylogenetic tree using phyML (http://www.phylogeny.fr/one\_task.cgi?task\_type=phyml) using the correctly formatted alignment from above as the input and specifying DNA/RNA; download output to be used with CodeML
   1. To download the file, click “Tree in Newick format” under “Outputs”
   2. The sample output can be seen in phyML\_output.trees
      1. Again, there are a few issues with formatting here that must be changed for CodeML the first of which being that there should not be new lines at all in the tree, so delete all new lines making sure that the entire tree is on one line (see the changes in formatted\_tree.trees)
      2. The second key detail is that the tree will not be accepted if it contains branch labels, so they must all be removed
         1. This can be done by removing all colons and numbers from the file (not including numbers in species names as defined above)
         2. Make sure to preserve the commas and parentheses structure of the file
3. Resulting files to be used with CodeML: .philip alignment file, .trees tree file (both properly formatted as specified above)

**PART 2: using CodeML in the terminal**

Inputs: .philip alignment file, .trees tree file, codeml.ctl control file, paml\_py.py

Outputs: results.out file, omega value

1. Type the following command in the terminal: pip install biopython
2. Install PAML locally (<http://abacus.gene.ucl.ac.uk/software/paml.html>) by following the instructions under the “UNIX/Linux and Mac OSX” section
   1. Download the .tgz file from the website and open a terminal window (command+spacebar and type terminal)
   2. In terminal, navigate to the “src” folder within the unzipped paml folder and type “make”. You will need to have xcode installed (or another working GCC compiler) in order for this to work.
   3. Once the compilation is complete, move the codeml executable to the “bin” folder within the unzipped paml folder (i.e., from the “src” folder, type “mv codeml ../bin/)
3. Edit the paml\_py.py file to reflect the names of your files and your local directory structure; edit the codeml.ctl file as desired
   1. More instructions can be found within the comments of the file itself
4. In the directory that contains your input files and paml\_py.py type: python3 paml\_py.py