Programming for Bioinformatics | BIOL 7200
November 27, 2018

This week we're going to do a bit of functional genomic analysis.

The only modules allowed for use in this exercise are: *os*, *sys*, *multiprocessing*, *subprocess*, and *argparse*

Instructions for submission

- **This assignment is due Monday, December 3, 2018 at 11:59pm. Late submissions will not be graded**
- Your code must be available on GitLab at the above time to be graded
- Add your code to `find_orthologs.py, blast_utilities/orthologs.py, blast_utilities/blast_wrapper.py`
- Your code should run as

```
./find_orthologs.py -i1 <Input file 1> -i2 <Input file 2> -o <Output
file name> –t <Sequence type – n/p>
```

- **DO NOT HARDCODE** any file name!

Exercise

Write a wrapper for finding orthologous genes using BLAST.

Identifying orthologous genes between different genomes is a very common task, and it's one of the, perhaps in fact the absolute, most important things that the comparative genomics group will do in the spring Computational Genomics class. There are a number of different ways of defining orthologous genes, but this week we are going to do it in a very simple way: reciprocal best BLAST hits. Reciprocal best BLAST hits are pairs of sequences where the best BLAST hit for each sequence is the other sequence.

Consider you have a sequence **A** from species s_A whose best hit in species s_B is the sequence **B**. **A** will be considered a reciprocal best hit of **B** if the best hit of **B** in species s_A is **A**.

Example: when you BLAST the complete set of human coding sequences against the complete set of mouse coding sequences, the best hit for the human gene histone H3.1 is the mouse gene histone H3.1 and vice versa. These two genes would be considered reciprocal best hits and orthologous. Please do note that this is an overly simplistic way of defining orthologous. If, for example, there had been a gene duplication event in mouse lineage yielding two copies of the histone H3 gene, then simply picking one as the ortholog for the human version would be a rather bad idea.

Your task for this exercise is to write a script that accomplishes the above example *en masse*. You will need the `makeblastdb` and the `blast` program. You can get them by installing the NCBI toolkit located here ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/LATEST. Your script should take in as arguments two sets of protein or nucleotide sequences, one from each genome. It should create a database from each using `makeblastdb`, query each set against the opposite database, and remove the databases files and any other temporary file that you created in the process. From the results of the queries, it should find those sequence pairs which are reciprocal best hits and give them as output.

Deliverables:

- *Code*: find_orthologs.py, blast_utilities/orthologs.py, blast_utilities/blast_wrapper.py
- *Output*: Your output file with orthologous genes named find_ortholog.output
- A README.txt describing how many initial blast hits you found and how many orthologous genes you found

Additional instructions:

- blast+ executables are assumed to be under your PATH
- Do not hard code the input file name