Pipeline manual

Elian Strozyk
elian.strozyk@etu.umontpellier.fr
elian.strozyk@gmail.com

20th July 2023

GitHub: https://github.com/elian-stz/StylED_internship

Contents

1 Introduction

This pipeline allows to **retrieve homologous sequences, infere orthology, align and make trees out of orthogroups** with a list of input NCBI identifiers. It also generates summary files for counting and make the information readily accessible. This pipeline uses **Snakemake** as a workflow management system. The directed acyclic graph is represented in figure 1.

- Input (more details in section 3):
  - A CSV file with a list of NCBI sequence identifiers
  - A BLAST algorithm version (`blastp`, `blastn`, `blastx`, or `tblastn`)
  - A taxonomic group (`Aphididae`, `Hemiptera`, `txid6656`, ...)
- Output (more details in section 4):
  - Fasta files (whole, per input identifier, and per species)
  - Aligned fasta files per orthogroup
  - Newick files per orthogroup (raw and leaf name tweaked)
  - Summary files as CSV and TXT

2 Getting started

2.1 Dependencies

You will need to download OrthoFinder in the `scripts/` directory, using `git clone` for example. Besides this, you will need to get sure that you have these tools and Python libraries:

- Unix-based OS
- An Internet connection
- Snakemake
- Python and the following libraries:
  - Biopython
  - Pandas
  - SciPy
  - NumPy
- OrthoFinder v2.5.5 (located in `./scripts/`)
- MAFFT
- IQ-TREE

2.2 Executing pipeline

Before running the pipeline, you need to get sure that all files are at the right location. Your working directory should contain these files: `Snakefile`, `config.yaml`, your CSV file; and the `scripts/` directory.

To run the pipeline, type:

```
snakemake -c1
```

You can set the number of cores by adjusting `-c` and the following number.

3 Input

The inputs are summarised in the `config.yaml` file. You will need to modify this file depending on your needs. Currently, the file looks like this:

```
csv_file: "stylin_NCBI_IDs.csv"
mode: "blastp"
organism: "Aphididae"
```

You will need to edit what is between quotes. Each argument is detailed below.

3.1 `csv_file` argument

This argument contains the CSV file located in your working directory. This file references the common name of the sequences, the NCBI identifiers of gene and protein. This file can contain as many as identifiers as you want.
If you want to create your own CSV file, you will **have to use the following header (without capital letters)**:

```
common_name,ncbi_gene_id,ncbi_protein_id
```

The pipeline provides an example CSV file named `stylin_NCBI_IDs.csv`, which looks like this:

common_name	**ncbi_gene_id**	**ncbi_protein_id**
Stylin-01	NM_001162314.1	NP_001155786.1
Stylin-02	NM_001162671.2	NP_001156143.1
Stylin-03	NM_001161959.2	NP_001155431.1
Stylin-04	NM_001172260.1	NP_001165731.1
Stylin-04bis	NM_001172268.2	NP_001165739.1
Stylin-05	NM_001163252.1	NP_001156724.1

Table 1: `stylin_NCBI_IDs.csv` provided in the pipeline. The attribute names in bold must be the same in your CSV file.

3.2 `mode` argument

For this argument, you need to precise the Blast algorithm version. It includes `blastp`, `blastn`, `blastx`, or `tblastn`. You need to type it without capital letters.
The tBlastx algorithm version is not implemented in the pipeline.

3.3 `organism` argument

This argument allows to limit the Blast search. You can type the taxonomic group, or the Ncbi taxonomy identifier[1] i.e. `txid` followed with the identifier, e.g. `txid7029` for *Acyrthosiphon pisum*.
You must provide a taxonomic group as a single word e.g. `Arthropoda`. Two-word taxa will not work. You will have to use `txid`, instead.

4 Output

This pipeline produces many files, here is the output and what the different directories contains (directories in red, files in blue).

```
(date)_(blast)_(organism)
 |__Fasta_per_input_ID
 |   |__Contains fasta files with homologous sequences per ID
 |__Fasta_raw
 |   |__Contains fasta files with all sequences
 |__Fasta_per_orthogroup
 |   |__Contains fasta files with homologous sequences per orthogroup
 |      (OrthoFinder output)
 |__Fasta_per_orthogroup_aligned
 |   |__Contains fasta files per orthogroup aligned with MAFFT
 |__IQ-TREE_output_per_orthogroup
 |   |__Contains IQ-TREE output per orthogroup
 |__Tree_per_orthogroup
 |   |__Contains Newick files per orthogroup, leaf names were adapted
 |      to be readable
 |__Sequence_number_per_species.txt
 |   |__Summary of sequence number per species
 |__Summary_per_orthogroup.csv
 |   |__Summary of fasta headers per orthogroup
 |__Summary_per_orthogroup_selected.csv
     |__Summary of fasta headers per orthogroup containing orthogroups
        for which an input ID was found
```

[1] Ncbi Taxonomy: https://www.ncbi.nlm.nih.gov/taxonomy

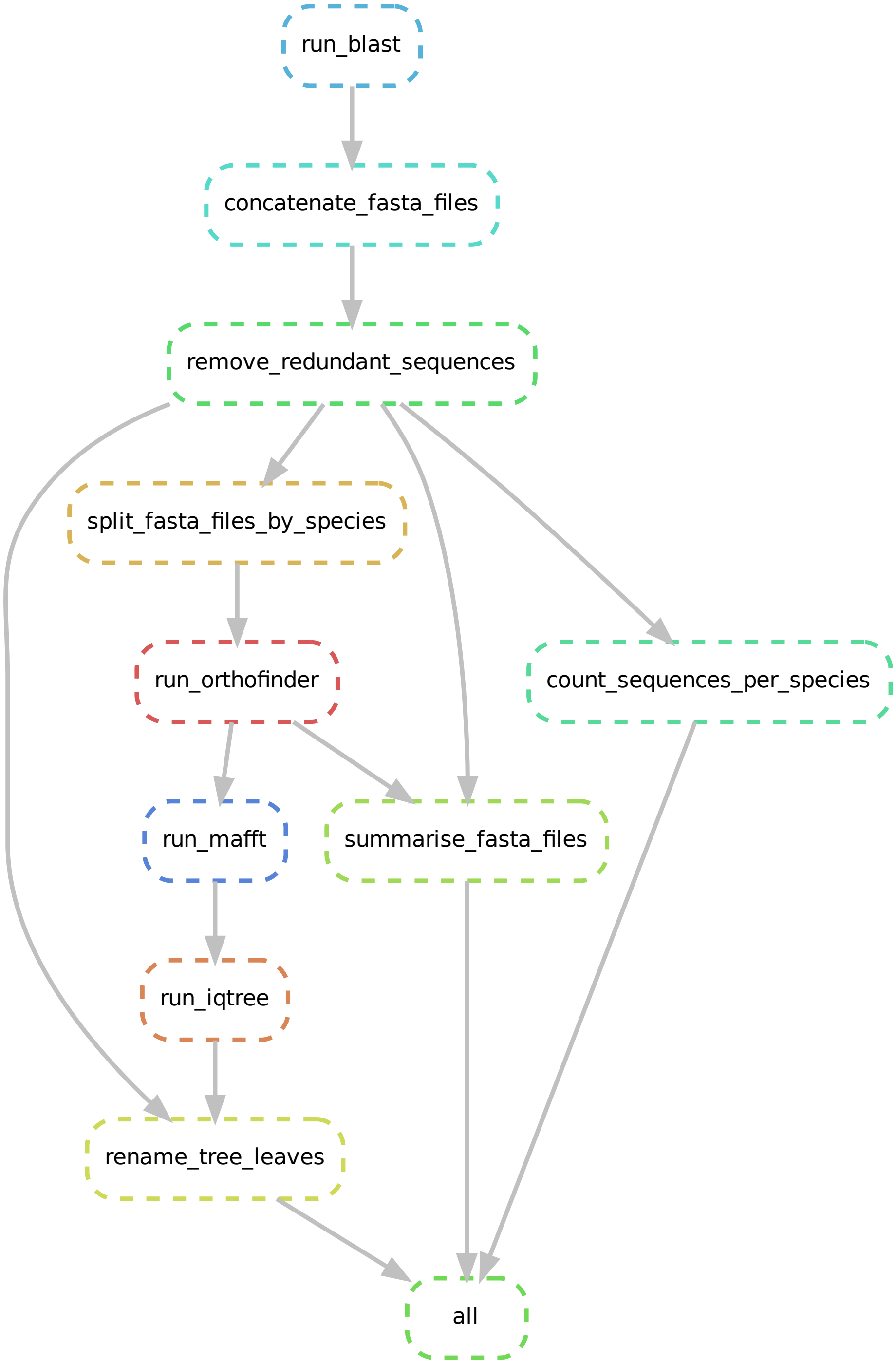

Figure 1: Directed acyclic graph of the workflow.