ShotgunUnifrac

Charlie Bushman

Jan 31, 2023

CONTENTS:

ShotgunUnifrac is a snakemake pipeline for generating a consensus phylogeny over many species. It can either create a phylogeny per core gene over a selection of genomes and then combining them together or it can append core genes directly into a super-alignment. The pipeline is written in snakemake and includes a python library for data curation and prep. View the *Quickstart Guide* page to see a worked example of installing and using the pipeline.

CHAPTER

ONE

QUICKSTART GUIDE

Contents

1.1 Installation

The first steps are to clone this repo and then install conda and snakemake. We can then install the Core Genes Extraction (CorGE) library and (optionally) run tests to make sure everything is properly installed.

```
git clone https://github.com/Ulthran/ShotgunUnifrac.git

cd ShotgunUnifrac/
pip install CorGE

pytest CorGE/tests/
pytest .tests/
```

Tip: If you've never installed Conda before, you'll need to add it to your shell's path. If you're running Bash (the most common terminal shell), the following command will add it to your path: `echo 'export PATH=$PATH:$HOME/miniconda3/bin' > ~/.bashrc`

If you see "Tests failed", file an issue on GitHub.

1.2 Setup

We'll start by creating some dummy inputs to work with using Escherichia coli, Buchnera aphidicola, Cellulomonas gilvus, Dictyoglomus thermophilum, and Methanobrevibacter smithii (a randomly selected group of gut bacteria plus Methanobrevibacter smithii as an outgroup for tree rooting).

```
echo $'7\n9\n2173' > EX_TXIDS.txt
echo $'GCF_000218545.1\nGCF_000020965.1' > EX_ACCS.txt
```

This creates two dummy input files

- `EX_TXIDS.txt` contains species-level taxon ids which CorGE will fetch genes and proteins for from NCBI.
- `EX_ACCS.txt` contains genome accessions which CorGE will fetch from NCBI.

Tip: You can curate genomes/proteins from NCBI using taxon ids and genome accessions but you can also (in the same command) gather local genomes/proteins as well using the `--local` flag.

1.2.1 Data curation

To gather the genes and proteins we need for tree building using the files we just created we use `CorGE collect_genomes`. Then to filter single copy core genes (SCCGs) from each file and merge nucleotide/amino acid sequences by SCCG we use `CorGE extract_genes`.

```
CorGE collect_genomes --ncbi_species EX_TXIDS.txt --ncbi_accessions EX_ACCS.txt
CorGE extract_genes
```

This should create the following directories and files from root

- ``output/``is a directory created to hold all of the below outputs
- `assembly_summary.txt` is downloaded from NCBI to find the best genome accessions for each taxon id.
- `config.yml` is provided to the snakemake pipeline to specify what it should look for and where.
- `genomes` is a directory containing each of the downloaded genomes (.faa and .fna)
- `filtered-sequences` is a directory containing each SCCG from each genome (protein-encoded) in their own files.
- `merged-sequences` is a directory containing each SCCG from each genome this time in per-SCCG files.

1.2.2 Tree building

You can check `output/config.yml` to see that the pipeline defaults to using the supermatrix approach to building a consensus phylogeny and rooting it with outgroup Methanobrevibacter smithii. Everything should be properly set up from running CorGE so we can just go ahead and run the pipeline.

```
snakemake all -c --use-conda --conda-prefix .snakemake/
```

This should create the following directories and files from root

- `RAxML_supermatrixRootedTree.final` is the final consensus tree.
- `aligned-sequences` is a directory containing alignments for the merged-sequences.

- `trees` is a directory containing phylogenies built from each SCCG alignment as well as some intermediates in the merging process.

Tip: `--use-conda` causes snakemake to use per-rule defined conda environments while it runs the pipeline. `--conda-prefix .snakemake/` tells conda where to put/look for these environments.

1.3 Viewing results

The output is `RAxML_supermatrixRootedTree.final` which can be viewed using any newick-format tree viewer (like ETE Toolkit).

1.4 tl;dr

Follow instructions to install anaconda / miniconda and snakemake then

```
git clone git@github.com:Ulthran/ShotgunUnifrac.git
cd ShotgunUnifrac/
echo $'7\n9\n2173' > EX_TXIDS.txt
echo $'GCF_000218545.1\nGCF_000020965.1' > EX_ACCS.txt
pip install CorGE
CorGE collect_genomes --ncbi_species EX_TXIDS.txt --ncbi_accessions EX_ACCS.txt
CorGE extract_genes
snakemake all -c --use-conda --conda-prefix .snakemake/
```

You should now have an output called `RAxML_supermatrixRootedTree.final`.

CHAPTER

TWO

CORGE USAGE GUIDE