SUMAC: Supermatrix Constructor version 1.0 Manual

William A. Freyman[1]

Department of Integrative Biology, University of California, Berkeley

[1] freyman@berkeley.edu

Contents

Chapter 1

Introduction

1.1 About SUMAC

SUMAC (Supermatrix Constructor) is a Python package to data mine GenBank and construct and evaluate phylogenetic supermatrices. It is designed to be run as a command-line program, though the modules can also be imported and used in other Python scripts. SUMAC will assemble supermatrices for any taxonomic group recognized in GenBank, and is optimized to run on multicore processors and clusters by utilizing multiple parallel processes.

When run from the command-line, SUMAC will perform a number of steps to create the phylogenetic supermatrix. First, SUMAC will download the GenBank database for the specified GenBank division (PLN, MAM, etc). SUMAC will then build clusters of homologous sequences in one of two ways: (1) perform exhaustive all-by-all BLAST comparisons of each ingroup and outgroup sequence and use a single-linkage hierarchical clustering algorithm, or (2) BLAST each ingroup and outgroup sequence against user provided guide sequences that define each cluster. SUMAC then discards clusters that are not phylogenetically informative (< 4 taxa), and then aligns each cluster of sequences using MUSCLE. Finally, the alignments are concatenated by species name (using the GenBank taxonomy) creating a supermatrix. A number of metrics are then calculated on the supermatrix, a graph indicating taxon coverage density is generated, and spreadsheets (in CSV format) are produced with information about each DNA region and GenBank accession used in the supermatrix. There are many options described in detail later in this manual.

1.2 Installation

1.2.1 Dependencies

The following dependencies must be installed to run SUMAC:

```
Python 2.7
Biopython
MUSCLE
BLAST+
```

The following are optional if you would like to generate graphs:

```
NumPy
matplotlib
```

1.2.2 Installing Dependencies on Linux

The following commands install the requirements for Debian GNU/Ubuntu Linux systems:

```
sudo pip install numpy
sudo pip install matplotlib
sudo pip install biopython
sudo apt-get install ncbi-blast+
sudo apt-get install muscle
```

TODO: test installation!!

1.2.3 Installing Requirements on Mac

TODO!

1.2.4 Installing SUMAC

TODO: test pip to install...

```
python setup.py install
```

1.3 License and Warranty

SUMAC is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

The program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details (http://www.gnu.org/copyleft/gpl.html).

Chapter 2

Quick Start Tutorial

This chapter provides the quick way to get started using SUMAC. There are many details described in Chapter 3 that would likely be helpful.

2.1 Construct a Supermatrix

The most basic usuage of SUMAC is to build a supermatrix with the following command:

```
python -m sumac -d pln -i Onagraceae -o Lythraceae
```

This command is an example of the minimum amount of input required to run SUMAC. The first part of the command `python -m sumac` runs the SUMAC module. The `-d pln` tell SUAMC to download the PLN (Plant) GenBank division. The `-i Onagraceae` and `-o Lythraceae` tells SUMAC to search the PLN division for all sequences within the taxonomic groups Onagraceae (the ingroup) and Lythraceae (the outgroup). SUMAC will then perform all-by-all BLAST comparisons of each sequence, build clusters of putatively homologous sequences, and construct a supermatrix.

Unless you are on a large multi-core system, the all-by-all BLAST comparisons will take a very long time to be performed since well over 5000 sequences will be found. To speed up the supermatrix construction, you could make a FASTA file of guide sequences to define each cluster. Each guide sequence could be an example of a sequence commonly used for phylogenetic analysis. You could then use this command:

```
python -m sumac -d pln -i Onagraceae -o Lythraceae -g guides.fasta
```

Which approach is better for constructing supermatrices? Using guide sequences makes supermatrix construction much faster, however it requires a priori knowledge of which DNA regions will be used in the supermatrix. Performing all-by-all BLAST comparisons is computationally more expensive, but it effectively data-mines GenBank in an exploratory fashion, so that sequence data not necessarily used in previous systematic studies can also be incorporated into the supermatrix. The decision will depend on the size of the taxonomic group being analyzed and the computational resources available.

2.2 Explanation of the Output Files

SUMAC will output the following files:

```
alignments/combined.fasta
alignments/N.fasta
clusters/N.fasta
gb_search_results
genbank_accessions.csv
gene_regions.csv
plot.pdf
```

```
sumac_log
```

The `alignments/combined.fasta` is the final aligned supermatrix in FASTA format. `alignments/N.fasta`, where N is an integer > 0, is the alignment of gene region N. Similarly, `clusters/N.fasta` is the unaligned raw sequence cluster of gene region N. The `gb_search_results` file is used by SUMAC to save the results of the GenBank sequence search in case the search is re-run. This file is not human readable.

The two CSV (comma-separated values) files contain tables that provide useful summary information about the supermatrix. `genbank_accessions.csv` is a table with each GenBank accession used, ordered by gene region and taxon (like the appendices found in most systematics papers). The `gene_regions.csv` file contains the number of taxa, the aligned length, the percent missing data, and the taxon coverage density of each gene region used in the supermatrix.

The `plot.pdf` file is a figure that shows how much sequence data was available for each taxon for each gene region. `sumac_log` is a log of the SUMAC run, and contains a great deal of information about the supermatrix construction, including final metrics such as the partical decisiveness (PD) of the supermatrix.

Chapter 3

SUMAC in Detail

3.1 Downloading GenBank

3.1.1 GenBank Division

The first time you run SUMAC you must specify which GenBank division to download with the `-d div` option, where `div` is the GenBank designated three letter code of the division (PLN, MAM, etc). Once SUMAC has downloaded the GenBank division, future SUMAC runs may leave out the `-d div` option to avoid repeatedly download the same files.

3.1.2 GenBank File Path

By default, each SUMAC run searches for the downloaded GenBank files in `./genbank/`, a subdirectory of the current run's directory. It may be useful to save the GenBank files outside of the current working directory, in which case you can specifiy the absolute path of the GenBank files with the `-p path` option. For example, if you want to build multiple supermatrices (or different versions of the same one) each in a different working directory it is helpful to use `-p /genbank` so that all SUMAC runs use the same copy of the GenBank files.

3.2 Specifying Ingroup and Outgroup

The `-i` and `-o` options must be used to specify which ingroup and outgroup to search for. The taxonomic names must be those used by GenBank. If a SUMAC run is repeated with the same ingroup and outgroup, SUMAC will load the previous search results to save time.

3.3 Using Guide Sequences

Guide sequences should be in a single standard FASTA file specified using the `-g` option. The names of the guide sequences will be ignored, and each of the ingroup and outgroup sequences will be BLASTed against the guide sequences.

3.4 Homologous Sequence Thresholds

3.4.1 BLAST E-value

By default, SUMAC uses a threshold default BLASTn e-value $1.0e-10$. This can be changed with the `-e` option.

3.4.2 Sequence Length Similarity

SUMAC uses a default threshold of sequence length percent similarity of 0.5. This can be changed with the `-l` option.

3.5 Partial Decisiveness

blah blah

3.6 Supermatrix Figure

blah blah