CSCI 5314
Algorithms in Molecular Biology

Spring 2016 Final Project Update

Group Members - QIIME	Group Members - mothur
Michael Oberg (**Grad**)	Mike Watson (**Grad**)
Johnathan Thompson	Ian Courtney
Chris Glass	Zane O'Brien
Matt Hong	Joseph Jackson
Charles Luckhardt	Jeffrey Stinemetze

Problem Domain

The Unifrac[1] algorithm is a distance metric for comparing $\boldsymbol{\beta}$-diversity in separate biological samples. $\boldsymbol{\beta}$-diversity as defined by Whittaker, et al (1972)[2] is "the extent of species replacement or biotic change along environmental gradients." This metric measures the distance between two sets of samples by determining the amount of evolutionary history that is shared. This is accomplished by comparing the phylogenetic tree branch lengths and uniqueness between the two sample sets. The Unifrac algorithm is implemented in both the QIIME[3] and mothur[4] analysis frameworks, in python and C++, respectively. There is also a weighted version of the Unifrac algorithm that takes into account relative abundances. The two variants of the Unifrac algorithm are typically referenced as *Qualitative* (unweighted) or *Quantitative* (weighted). In most analysis, both methods would be deployed. Unweighted Unifrac is the dominant metric when the difference between the samples can be characterized by one or more environmental elements (such as temperature), where abundance information obscures significant patterns of variation in a given taxonomy. Weighted Unifrac (a *Quantitative* measure) helps to reveal differences when there is relative abundance due to a factor such as the availability of a rate-limiting nutrient.

[1] Lozupone, Catherine, and Rob Knight. "UniFrac: a new phylogenetic method for comparing microbial communities." *Applied and environmental microbiology* 71.12 (2005): 8228-8235.

[2] Whittaker, Robert H. "Evolution and measurement of species diversity." *Taxon* (1972): 213-251.

[3] Caporaso, J Gregory et al. "QIIME allows analysis of high-throughput community sequencing data." *Nature methods* 7.5 (2010): 335-336.

[4] Schloss, Patrick D et al. "Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities." *Applied and environmental microbiology* 75.23 (2009): 7537-7541.

Current Status

As of 04/14/2016, our teams have completed the following tasks:

- Created three github repositories:
  - *unifrac-common* (grinder, benchmarking, writeups)
  - *unifrac-qiime* (qiime C re-implementation)
  - *unifrac-mothur* (mothur C re-implementation)
- Created *unifrac.slack.com* for group discussions
  - Channels for *general, benchmarking, grinder, mothur, qiime*
- Resurrected physical HPC cluster for dedicated and high-performance resources:
  - Restored two old admin nodes (required fair amount of work as both were broken by previous PhD student, i.e. one was booted into a live CD for recovery)
  - Restored NFS file servers (~20TB cluster-wide storage)
  - Restored 40-core, 1TB RAM analysis node (*dav01*)
    - Secured, configured new external IP
    - Created accounts for all team members
- Initial system-wide software installation and development on *dav01*:
  - Grinder installed and tested
  - GreenGenes database installed and uncompressed (over 50GB)
  - Installed mothur system-wide
  - Resurrected older version of QIIME (1.7.0) and 23 additional support packages (blast_2.2.22, biom-format_1.1.2, cdbtools_, cython_0.19.1, cytoscape_2.7.0, fasttree_2.1.3, elementtree_1.2.6, gdata_2.0.17, gsl_1.9, infernal_1.0.2, matplotlib_1.1.0, mpi4py_1.2.2, muscle_3.8.31, numpy_1.5.1, pycogent_1.5.3, pynast_1.2, python_2.7.3, R_2.15.2, tax2tree_v1.0, uclust_1.2.22, usearch_5.2.236, usearch_6.1.544, scipy_0.12.0)
- Grinder research and testing
  - Initial fasta file created from GreenGenes (5 species, 10 lines)
  - Initial Grinder tests against this fasta file: 100 reads, 300 lines
- Reviewed Human Microbiome Project for species found in specific human microbiomes (skin, bone) to begin initial selection of species for our analysis

Grinder

We identified several example grinder executions[5], are in the process of understanding and evaluating these options to develop a canonical grinder simulation that maps to CU's MiSeq[6]:

```
./grinder -rf reference_genomes -af hc.txt -tr reads_number -rd 250 -md linear 1 2 –fq 1 –ql 30 10 –hd Balzer –bn Grinder
./grinder -rf reference_genomes -af hc.txt -tr reads_number -rd 75 -md poly4 3e-3 3.3e-8 –fq 1 –ql 30 10 –bn Grinder
./grinder -rf reference_genomes -af percentage.txt -tr 475,694 -mr 17 83 -md linear 1 2 -rd 193 normal 60 –fq 1 –ql 30 10 –hd Balzer –bn Grinder-DatasetE-self
./grinder -rf reference_genomes -af percentage.txt -tr 2,975,345 -rd 75 -md poly4 3e-3 3.3e-8 -mr 95 5 –fq 1 –ql 30 10 –bn Grinder
```

Benchmarking

Methods to Generate Test Data

The Grinder sample simulator will be used to generate simulated samples from the GreenGenes, 16S rRNA gene database. Developing simulated sample sets for this project from a known, canonical set will allow us to process these samples and benchmark the target Unifrac algorithm, while retaining the ability to compare our analysis pipeline results back to a reference sequence database.

Species Selection

We are looking at how to best select species to develop two individual fasta files, representing two "samples", with a known UniFrac distance (0.5 to start). We are also beginning to identify the correct inputs to obtain suitable weighted UniFrac results.

The next major step in our benchmarking effort is to identify the areas of the analysis pipeline that we need to benchmark in order to correctly identify any variance in the UniFrac algorithm. We are still looking at the QIIME and mothur pipelines to identify the series of steps required to provide the necessary inputs to the UniFrac components of each. For QIIME, the UniFrac

[5] "commands." 2013. 14 Apr. 2016 <http://cbb.sjtu.edu.cn/~ccwei/pub/software/NeSSM/command.pdf>

[6] "Next-Generation Sequencing Facility - BioFrontiers Institute." 2012. 14 Apr. 2016 <https://biofrontiers.colorado.edu/core-facilities/next-gen-sequencing>

portion takes an OTU table and phylogenetic tree, and mothur takes a phylogenetic tree and the number of iterations to be performed. For mothur, there are options for group, name, count, distance, groups, iterations, processors, root and random - and we are still analyzing the analysis pipeline and code to identify the use and suitability of each of these options for our project.

We are beginning to develop the python matplotlib scripting infrastructure in order to plot and report these results in a publication ready form.

Test Cases to Test Algorithm Limits

The development of the simulated samples using Grinder, which will then be run and benchmarked through our two parallel pipelines included the development of test cases that highlight the strengths and weaknesses of the Unifrac algorithm. These test cases will be developed against the QIIME and mothur implementations of the Unifrac algorithm, and will be used to test the efficacy and suitability of the re-implemented versions.

Algorithm Re-implementation

The Unifrac implementation included in the QIIME suite (implemented as *beta_diversity.py*) is written completely in Python, and the implementation included in mothur is in C++. For this project we will re-implement both of these in optimized C, and compare to the original versions.

This development process will include an analysis of the original performance of the implementations (in Python and C++), and comparison to the re-written versions for each analysis framework.

Division of Work and Milestones

Principal means this person has primary responsibility. ***Partner***(s) will support Principal as needed. These are preliminary assignments that will need to be revised by the respective teams and team members.

QIIME:

Project Deliverable	QIIME Principal	QIIME Partner(s)	Date
Grinder Simulated Sample Creation	Matt Hong	Charles Luckhardt	04/12/16
Benchmark Environment	Chris Glass	Michael Oberg / Johnathan Thompson	04/12/16
Unifrac Re-implementation	Johnathan Thompson	Chris Glass / Michael Oberg	04/21/16
Project Writeup	Michael Oberg	Matt Hong / Charles Luckhardt	04/28/16

mothur:

Project Deliverable	mothur Principal	mothur Partner(s)	Date
Grinder Simulated Sample Creation	Ian Courtney	Zane O’Brien	04/12/16
Benchmark Environment	Jeffrey Stinemetze	Mike Watson / Joseph Jackson	04/12/16
Unifrac Re-implementation	Joseph Jackson	Jeffrey Stinemetze / Mike Watson	04/21/16
Project Writeup	Mike Watson	Ian Courtney / Zane O’Brien	04/28/16

Application Sources

https://github.com/biocore/qiime/blob/master/qiime
https://github.com/mothur/mothur

References

1. Whittaker, Robert H. "Evolution and measurement of species diversity." *Taxon* (1972): 213-251.
2. Lozupone, Catherine, and Rob Knight. "UniFrac: a new phylogenetic method for comparing microbial communities." *Applied and environmental microbiology* 71.12 (2005): 8228-8235.
3. Caporaso, J Gregory et al. "QIIME allows analysis of high-throughput community sequencing data." *Nature methods* 7.5 (2010): 335-336.
4. "beta_diversity.py – Calculate beta diversity (pairwise ... - Qiime." 2011. 24 Mar. 2016 <http://qiime.org/scripts/beta_diversity.html>
5. Schloss, Patrick D et al. "Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities." *Applied and environmental microbiology* 75.23 (2009): 7537-7541.
6. "mothur." 2009. 26 Mar. 2016 <http://www.mothur.org/>
7. https://www.researchgate.net/post/What_is_the_difference_between_alpha-diversity_and_beta-diversity_in_microbial_ecology
8. Oberg, Michael et al. "A system architecture supporting high-performance and cloud computing in an academic consortium environment." *Computer Science-Research and Development* 26.3-4 (2011): 317-324.