MolBiol 3I03 Midterm Progress Report

Siddharth Reed

February 22, 2020

1 Writing

I have trimmed down my submitted thesis significantly closer to the size of a publication. Most of the cutting was done from the introduction section, removing lots of unnecessary background information, going from 11 to 3 pages. More may still be cut, especially to make room to discuss the failings of Gophna et al and why this paper uses more statistically rigorous and sensitive methods. I also rewrote what was there to be more concise and clear, again removing lines or details that were not particularly relevant to the methods or result. I updated the methods to state that we now use whole genome alignment to build the species trees, not 16S genes. Currently the results and discussion sections are unchanged from the thesis as I will be regenerating the results with more data and updating some of the figures using ggplot instead of matplotlib.

2 Code

The repository is at `https://github.com/DJSiddharthVader/thesis_SidReed`

2.1 Scraping Data

Since I switched to using whole genome alignments to build the species trees I need to scrape whole genomes from NCBI. I added this functionality to the existing scripts I had written to scrape the nucleotide and protein data in the "scripts/scrapingData" folder of the repository. It was mostly re-writing the scripts to be more general and then adding the arguments specific to the genome files that needed to be downloaded I also added comments and rewrote some lines to make the scripts more legible to people looking at the code.

2.2 Whole Genome Species Trees

The other major thing was writing the script to build the whole genome alignment. This was done since for many genera considered preciously I could not build species trees from 16S rRNA as there were not enough genes annotated for each organism in the genus of interest. A whole genome alignment completely bypasses this need for annotated 16S genes and allows us to use much more information to build the species tree. First I investigated several whole-genome alingers, namely Mugsy, Mauve and Parsnp. All of these programs take in fasta files and output multiple alignments that can be parsed using biopython (either maf or xmfa format). Mugsy I was able to get running on infoserv but it only seems to work when it was run in the same directory as the genome files used for input which is annoying for running it in a python script. It was relatively quick when run on 5 genomes ($\approx$ 10 minutes) but I do not know how well it scales. Mauve, specifically progressiveMauve, was also easily installed and run on infoserv. It is also supposed to be significantly more accurate due to its iterative refinement but it is also considerably slower ($\geq$ 2 hours on the same 5 genomes) and did not appear scale well ($\geq$ 2 days on 100 genome). Parsnp is slightly faster than mugsy and is easier

to specify the input and output files. Parsnp also does core genome alignment, not whole genome alignment which is appropriate in this case since I only want to use the alignment to build a species tree. Largely due to ease of use I am currently using parsnp to generate the genome alignments but this may change in the future.

Given a multi alignment (xmfa format) from parsnp there is some preprocessing required before I can pass the data to MrBayes which requires a single nexus file as input. XMFA files are a set of separate multiple alignments separated by lines of = characters, thus they can be easily split up into a set of separate multialignment fasta files. These files can each be individually converted to nexus files and then concatenated together into a single nexus file using the biopython nexus modules. Further some of these aligned are filtered out as they are uninformative, meaning the sequence is exactly the same for every organism in the alignment. It is necessary to filter these uninformative regions as MrBayes has a limit on the total number of characters it will accept for an alignment.

Otherwise I have been adding comments and doing minor rewrites of other scripts for legibilty while testing that they still work.

3 Future Work

A list of things to finish

- Change model matrix from BDSYM to BDARD for the markophylo analysis
  - Warning when using BDSYM or BDARD matrix and numhessian TRUE or FALSE and alphabet $(0, 1)$ is "Estimated parameters on interval bounds."
  - Error when using BDSYM or BDARD matrix and numhesian TRUE or FALSE with the alphabet in $(1, 2)$

```
Something is not right with the standard errors.Check Hessian matrix
    estimate.
Consider calculating bootstrap errors (make sure to use numhessian=
    FALSE).
Warning message:
In nlminb(start = modelop[[i]]$start, objective = totalll, model = i,  :
  NA/NaN function evaluation
```

- Rerun the analysis with the updated data and likely increase the sampling size (samples more than 50 trees per bootstrap, create more than 1500 gene trees to sample from)
- Update the figures
  - Remake the figures with the new data pulled from NCBI
  - Likely remake the using R (ggplot2) as they will look nicer and simplify the code a lot. ggplot will also allow faster tweaking/prototyping of figures.
- Rewrite and edit the results and discussion sections
- Continue adding comments the remaining scripts and the README.md
- Look into depositing the data used (the raw network data, the markophylo estimates and all the species and gene trees) on Zenodo, but this can wait until everything else is finished