README.md

PIrANHA

PhylogenetIcs ANd pHylogeogrAphy

codacy A license GPL (>= 2)

Scripts for file processing and analysis in phylogenomics & phylogeography

LICENSE

CITATION

If you use scripts from this repository as part of your published research, I require that you cite the repository as follows (also see DOI information below):

- Bagley, J.C. 2017. PIrANHA. GitHub repository, Available at: http://github.com/justincbagley/PIrANHA.

Alternatively, please provide the following link to this software repository in your manuscript:

- https://github.com/justincbagley/PIrANHA

DOI

The DOI for `PIrANHA`, via Zenodo, is as follows: DOI 10.5281/zenodo.890754. Here are some examples of citing PIrANHA using the DOI:

Bagley, J.C. 2017. PIrANHA. GitHub package, Available at: http://doi.org/10.5281/zenodo.846401.

Bagley, J.C. 2017. PIrANHA. Zenodo, Available at: http://doi.org/10.5281/zenodo.846401.

INTRODUCTION

Taking steps towards automating boring stuff during analyses of genetic data in phylogenomics & phylogeography...

`PIrANHA` v0.1.6.1 is a repository of shell scripts and R scripts written by the author, as well as additional code (R, Perl, and Python scripts) from other authors, that is designed to help automate processing and analysis of DNA sequence data in phylogenetics and phylogeography research projects (Avise 2000; Felsensetin 2004). PIrANHA is fully command line-based and, rather than being structured as a single pipeline, it contains a series of scripts, some of which form pipelines, for aiding or completing tasks during evolutionary analyses of genetic data. Currently, PIrANHA scripts facilitate running or linking the following software programs:

- **`pyRAD`** (Eaton 2014) or **`ipyrad`** (Eaton and Overcast 2016)
- **`PartitionFinder`** (Lanfear et al. 2012, 2016)

- `BEAST` (Drummond et al. 2012; Bouckaert et al. 2014)
- `starBEAST` (Heled & Drummond 2010)
- `SNAPP` (Bryant et al. 2012)
- `MrBayes` (Ronquist et al. 2012)
- `ExaBayes` (Aberer et al. 2014)
- `RAxML` (Stamatakis 2014)
- `dadi` (Gutenkunst et al. 2009)
- `fastSTRUCTURE` (Raj et al. 2014)
- `PhyloMapper` (Lemmon and Lemmon 2008)

The current code in `PIrANHA` has been written largely with a focus on 1) analyses of DNA sequence data and SNPs or SNP loci generated from massively parallel sequencing runs on ddRAD-seq genomic libraries (e.g. Peterson et al. 2012), and 2) automating running these software programs on the user's personal machine (e.g. MAGNET pipeline and pyRAD2PartitionFinder scripts) or a remote supercomputer machine, and then conducting post-processing of the results. In particular, a number of scripts have been written with sections allowing them to be run (or cause other software to be called) on a supercomputing cluster, using code suitable for SLURM or TORQUE (PBS; Portable Batch System) resource management systems (in some cases, this functionality is noted by adding "Super" in the script filename, as in `Super-pyRAD2PartitionFinder.sh`).

Distribution Structure and Pipelines

What's new in this release?

The current release, `PIrANHA` v0.1.6.1, updates the README and documentation for the repository.

The previous release, `PIrANHA` v0.1.6, added the following updates, in addition to minor improvements in the code:

- **August 2017:** + added a Change Log file (`'changeLog.md'`) to supplement releases page and provide log file within master.
- **August 2017:** + updated `MAGNET` pipeline by editing `'MAGNET.sh'` by adding three new command line options ("-e", "-m", and "-o" flags), as follows: `\-e executable (def: raxmlHPC-SSE3) name of` RAxML `executable, accessible from command line on user's machine \-m indivMissingData (def: 1=allowed; 0=removed) \-o outgroup (def: NULL) outgroup given as single taxon name (tip label) or comma-separted list`
- **August 2017:** + updated `MAGNET` pipeline by adding `getBootTrees.sh` script, which collates and organizes bootstrap trees from all RAxML runs in sub-folders of a working directory, especially results of a MAGNET run. This is the standalone version of the script.
- **August 2017:** + updated `'BEAST\_PSPrepper.sh'` script automating editing existing `BEAST` v2+ (e.g. v2.4.5) input XML files for path sampling analysis, so that users don't have to do this by hand!
- **2017** + added `'phyNcharSumm.sh'` script in `MAGNET/shell` dir, which creates table summarizing the number of characters (length, in bp) in each of multiple PHYLIP sequence alignments in a working directory.
- **2017** + added `'phyNcharSumm.sh'` script in `MAGNET/shell` dir, which creates table summarizing the number of characters (length, in bp) in each of multiple PHYLIP sequence alignments in a working directory.
- **bug fix:** - `MAGNET.sh` (unused code)
- **bug fix:** - `getGeneTrees.sh` (unused code)
- **bug fix:** - `BEASTRunner.sh`

What is possible with PIrANHA? Who cares?

How PIrANHA scripts work together

The goal of `PIrANHA` is to facilitate analysis pipelines that could be of interest to nearly anyone conducting evolutionary analyses of DNA sequence data using maximum-likelihood and Bayesian methods. **Figure 1** and **Figure 2** below demonstrate flow and interactions of the current partition scheme, population structure, and phylogenetics pipelines with **software** and **"file types"** used to generate input for PIrANHA in the left column, and the way these are processed within/using `PIrANHA` illustrated in the right column. External software programs are shown in balloons with names in black italic font, while `PIrANHA` scripts are given in blue. Arrows show the flow of files through different pipelines, which terminate in results (shown right of final arrows at far right of each diagram).

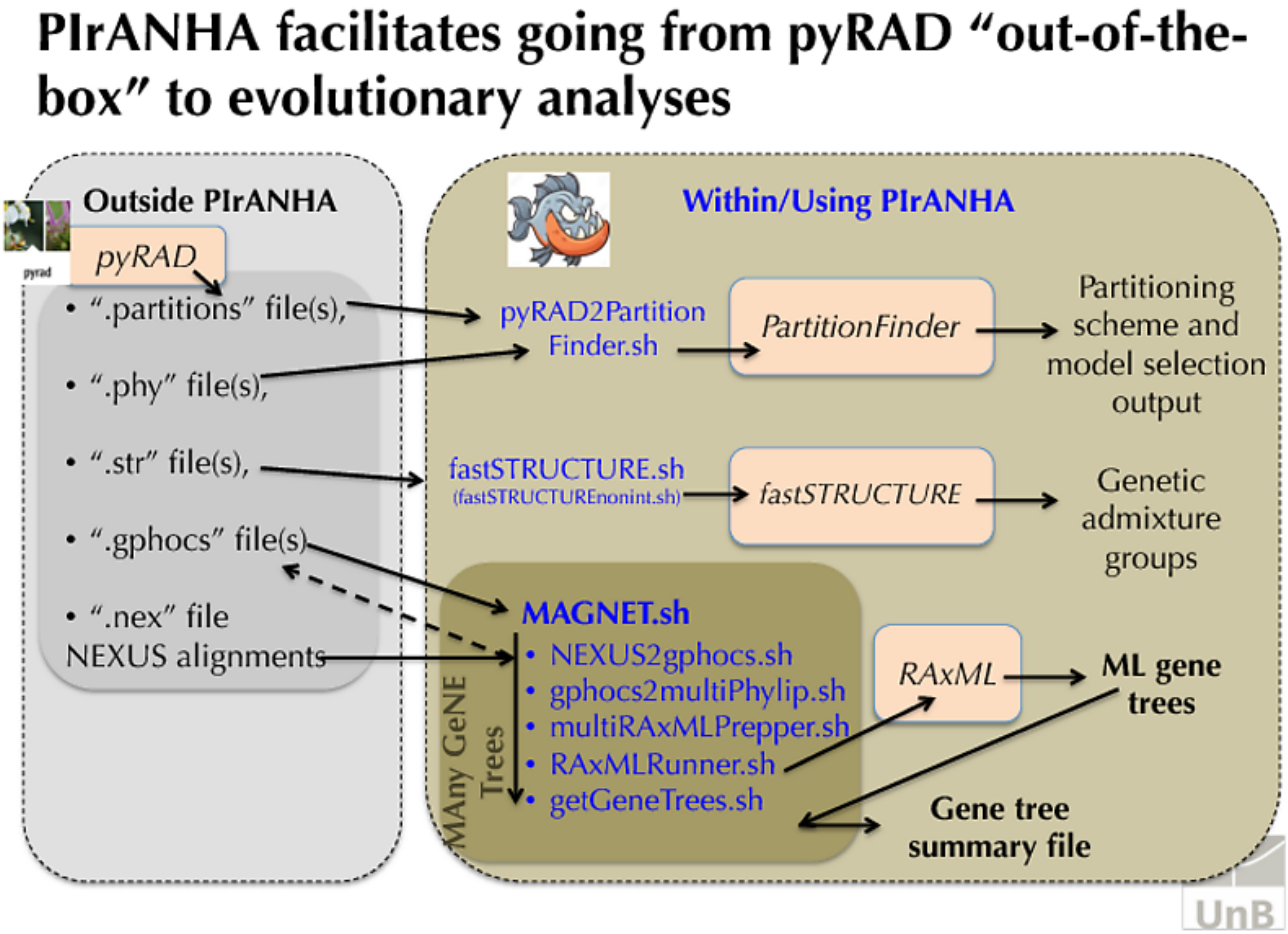

Figure 1

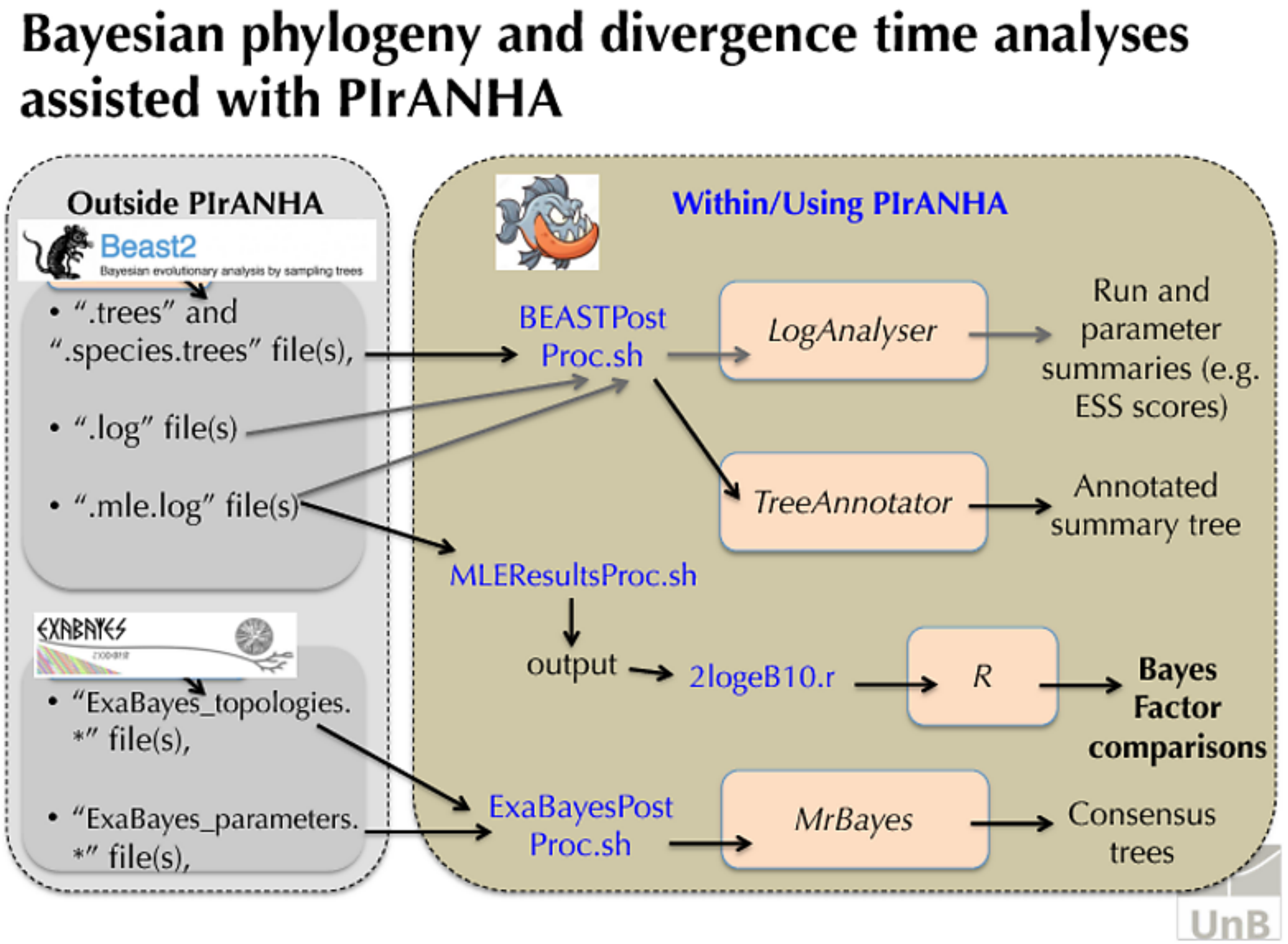
Bayesian phylogeny and divergence time analyses assisted with PIrANHA
Outside PIrANHA
Beast2
Bayesian evolutionary analysis by sampling trees
• ".trees" and ".species.trees" file(s),
• ".log" file(s)
• ".mle.log" file(s)
EXABAYES
• "ExaBayes_topologies.*" file(s),
• "ExaBayes_parameters.*" file(s),
Within/Using PIrANHA
BEASTPost Proc.sh
LogAnalyser
Run and parameter summaries (e.g. ESS scores)
TreeAnnotator
Annotated summary tree
MLEResultsProc.sh
output
2logeB10.r
R
Bayes Factor comparisons
ExaBayesPost Proc.sh
MrBayes
Consensus trees
UnB

Figure 2

The following **Figure 3** illustrates new capacities of running and processing dadi (Gutenkunst et al. 2009) files in `PIrANHA` (note: the post-processing script is still under development).

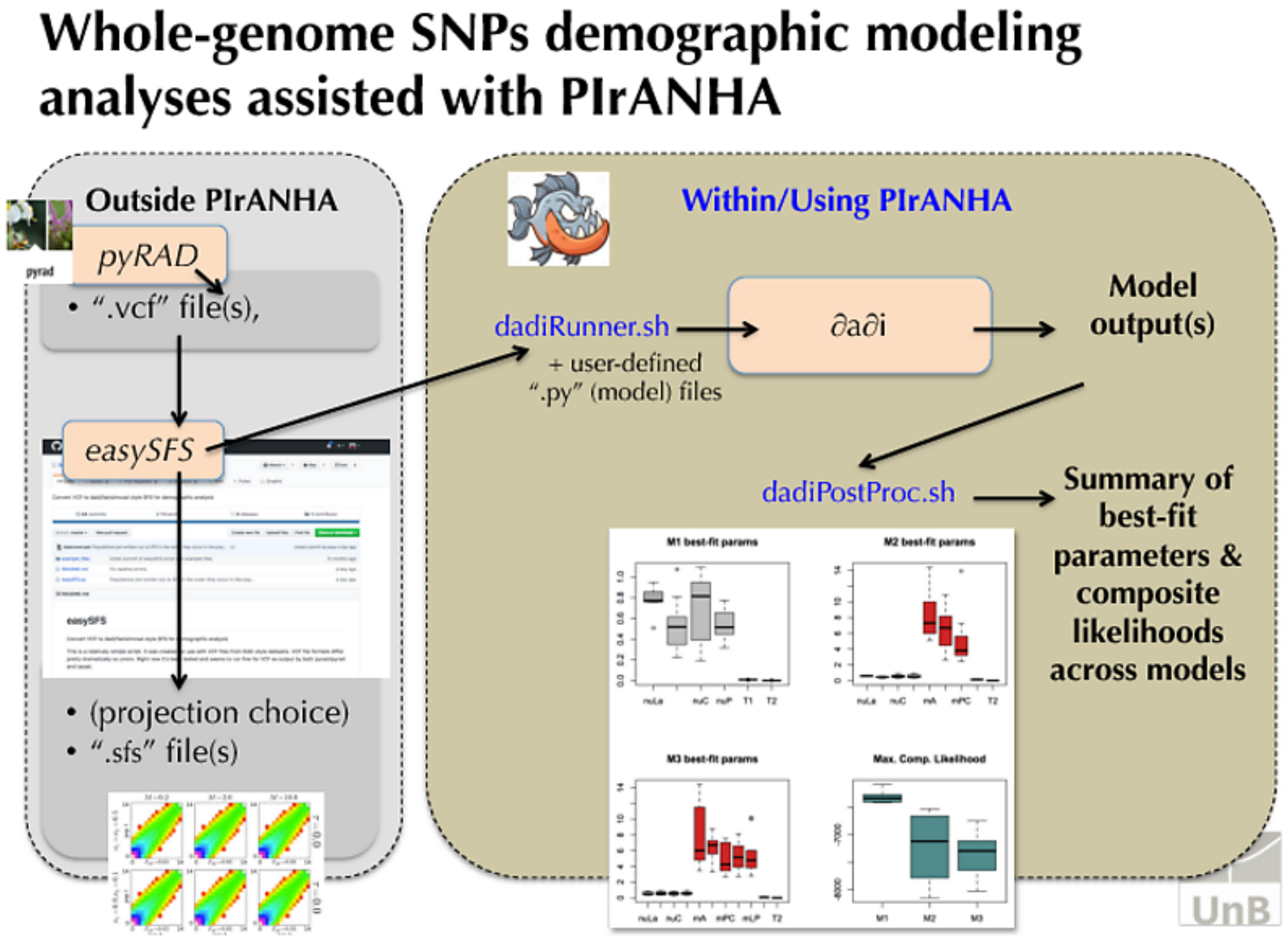

Figure 3

GETTING STARTED

Dependencies

`PIrANHA`, and especially the `MAGNET` package (here or here) within `PIrANHA`, relies on several software dependencies. These dependencies are described in some detail in README files for different scripts or packages; however, I provide a full list of them below, with asterisk marks preceding those already included in the `MAGNET` subdirectory of the current release. Of course, you can usually get away with not installing dependencies or software that are unrelated to the analysis you are conducting with `PIrANHA`, but it is recommended that you install all dependencies to take full advantage of PIrANHA's capabilities, or be prepared for any analysis!

- `PartitionFinder`
- `BEAST` v1.8.3++ and v2.4.2++ (or newer; available at: http://beast.community and http://beast2.org, respectively)
  - Updated `Java`, appropriate Java virtual machine / jdk required
  - `BEAGLE` in beagle-lib (`libhmsbeagle`* files) required
  - default `BEAST` packages required
  - `SNAPP` package addon required
- `MrBayes` v3.2++ (available at: http://mrbayes.sourceforge.net/download.php)
- `ExaBayes` (available at: http://sco.h-its.org/exelixis/web/software/exabayes/)
- `RAxML` (available at: http://sco.h-its.org/exelixis/web/software/raxml/index.html)
- `Perl` (available at: https://www.perl.org/get.html).

- *Nayoki Takebayashi's file conversion Perl scripts (available at: http://raven.iab.alaska.edu/~ntakebay/teaching/programming/perl-scripts/perl-scripts.html; note: some, but not all of these, come packaged within MAGNET)
- `Python` v2.7 and/or 3++ (available at: https://www.python.org/downloads/)
  - `Numpy` (available at: http://www.numpy.org/)
  - `Scipy` (available at: http://www.scipy.org/)
  - `Cython` (available at: http://cython.org/)
  - GNU Scientific Library (available at: http://www.gnu.org/software/gsl/)
  - `bioscripts.convert` v0.4 `Python` package (available at: https://pypi.python.org/pypi/bioscripts.convert/0.4; also see README for `'NEXUS2gphocs.sh'`)
- `fastSTRUCTURE` v1.0 (available at: https://rajanil.github.io/fastStructure/)
- `dadi` v1.7.0++ (or v1.6.3; available at: https://bitbucket.org/gutenkunstlab/dadi/overview)
- `R` v3++ (available at: https://cran.r-project.org/)

Users must install all software not included in `PIrANHA`, and ensure that it is available via the command line on their supercomputer and/or local machine (best practice is to simply install all software in both places). For more details, see the `MAGNET` README.

Installation

💻 As `PIrANHA` is primarily composed of UNIX shell scripts and customized R scripts, it is well suited for running on a variety of types of machines, especially UNIX/LINUX-like systems that are now commonplace in personal computing and dedicated supercomputer cluster facilities. The UNIX shell is common to all Linux systems and mac OS X. There is no installation protocol for `PIrANHA`, because these systems come with the shell preinstalled; thus `PIrANHA` should run "out-of-the-box" from most any folder on your machine.

IMPORTANT! - Passwordless SSH Access

`PIrANHA` largely focuses on allowing users with access to a remote supercomputing cluster to take advantage of that resource in an automated fashion. Thus, it is implicitly assumed in most scripts and documentation that the user has set up passowordless ssh access to a supercomputer account.

✋ If you have not done this, or are unsure about this, then you should set up passwordless acces by creating and organizing appropriate and secure public and private ssh keys on your machine and the remote supercomputer prior to using PIrANHA. By "secure," I mean that, during this process, you should have closed write privledges to authorized keys by typing "chmod u-w authorized keys" after setting things up using ssh-keygen.

❗ Setting up passwordless SSH access is **VERY IMPORTANT** as `PIrANHA` scripts and pipelines will not work without setting this up first. The following links provide a list of useful tutorials/discussions that can help users set up passwordless SSH access:

- https://www.msi.umn.edu/support/faq/how-do-i-setup-ssh-keys
- https://coolestguidesontheplanet.com/make-passwordless-ssh-connection-osx-10-9-mavericks-linux/
- https://www.tecmint.com/ssh-passwordless-login-using-ssh-keygen-in-5-easy-steps/

Input and Output File Formats

📄 `PIrANHA` scripts accept a number of different input file types, which are listed in Table 1 below. These can be generated by hand or are output by specific upstream software programs. As far as *output file types* go, `PIrANHA` outputs various text, PDF, and other kinds of graphical output from software that are linked through `PIrANHA` pipelines.

Input file types	Software (from)

.partitions	`pyRAD` / `ipyrad`
.phy	`pyRAD` / `ipyrad` / by hand
.str	`pyRAD` / `ipyrad`
.gphocs	`pyRAD` / `ipyrad` / `MAGNET` (`NEXUS2gphocs.sh`)
.loci	`pyRAD` / `ipyrad`
.nex	`pyRAD` / `ipyrad` / by hand
.trees	`BEAST`
.species.trees	`BEAST`
.log	`BEAST`
.mle.log	`BEAST`
.xml	`BEAUti`
.sfs	`easySFS`
Exabayes_topologies.*	`ExaBayes`
Exabayes_parameters.*	`ExaBayes`

🚧 *NOTE: The following 'Getting Started' content is Under Construction! E-mail me about it, or check back soon for updates!* 🚧

Phylogenetic Partitioning Scheme/Model Selection

pyRAD2PartitionFinder

Shell script for going directly from `PHYLIP` alignment (.phy) and partitions (.partisions) files output by `pyRAD` (Eaton 2014) or `ipyrad` (Eaton and Overcast 2016; for de novo assembly of reduced-representation sequence data from an NGS experiment) to inference of the optimal partitioning scheme and models of DNA sequence evolution for `pyRAD` -defined SNP loci in PartitionFinder (Lanfear et al. 2012, 2016). See current release of `pyRAD2PartitionFinder` scripts for more information (e.g. detailed comments located within the code itself; a README is coming soon).

Estimating Gene Trees for Species Tree Inference

MAGNET (MAny GeNE Trees)

Interactive shell pipeline for inferring maximum-likelihood gene trees in `RAxML` (Stamatakis 2014) for multilocus DNA sequence alignments (e.g. RAD loci from ddRAD-seq experiments, candidate genes, genomic contigs) to aid downstream summary-statistics species tree inference. Please see the README for the `MAGNET` Package, which is available as its own stand-alone repository so that it can be tracked and continually given its own updated doi and citation by Zenodo. Three starting input file formats are currently supported, including single NEXUS (.nex), single `G-PhoCS` (.gphocs; formatted for `G-PhoCS` software, Gronau et al. 2011), and multiple `PHYLIP` files.

Automating Bayesian evolutionary analyses in BEAST

BEASTRunner

BEASTRunner automates conducting multiple runs of `BEAST` v1 or v2 (Drummond et al. 2012; Bouckaert et al. 2014) XML input files on a remote supercomputing cluster that uses SLURM resource management with PBS wrappers, or a TORQUE/PBS resource management system. See the `BEASTRunner` README for more information.

BEAST_PathSampling

The BEAST_PathSampling directory is a new area of development within PIrANHA in which I am actively coding scripts to (1) edit `BEAST` v2++ XML files for path sampling analyses (Xie et al. 2011; Baele et al. 2012) and (2) automate moving/running the new path sampling XML files on a supercomputing cluster. Even as of August 2017, this is*very new stuff that is experimental and may still not be working*, so stay tuned for more updates soon.

ACKNOWLEDGEMENTS

I thank Nayoki Takebayashi, who wrote and freely provided some Perl scripts I have used in `PIrANHA`. I also thank the Brigham Young University Fulton Supercomputing Lab (FSL) for providing computational resources used during the development of this software. J.C.B. received stipend support from a Ciência Sem Fronteiras (Science Without Borders) postdoctoral fellowship from the Brazilian Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq; Processo 314724/2014-1). Lab and computer space was also supplied by The University of Alabama, during an internship in the Lozier Lab in the UA Department of Biological Sciences.

REFERENCES

- Aberer AJ, Kobert K, Stamatakis A (2014) ExaBayes: massively parallel Bayesian tree inference for the whole-genome era. Molecular Biology and Evolution, 31, 2553-2556.
- Avise JC (2000) Phylogeography: the history and formation of species. Cambridge, MA: Harvard University Press.
- Baele G, Lemey P, Bedford T, Rambaut A, Suchard MA, Alekseyenko AV (2012) Improving the accuracy of demographic and molecular clock model comparison while accommodating phylogenetic uncertainty. Molecular Biology and Evolution, 29, 2157-2167.
- Bouckaert R, Heled J, Künert D, Vaughan TG, Wu CH, Xie D, Suchard MA, Rambaut A, Drummond AJ (2014) BEAST2: a software platform for Bayesian evolutionary analysis. PLoS Computational Biology, 10, e1003537.
- Bryant D, Bouckaert R, Felsenstein J, Rosenberg NA, RoyChoudhury A (2012) Inferring species trees directly from biallelic genetic markers: bypassing gene trees in a full coalescent analysis. Molecular Biology and Evolution, 29, 1917–1932.
- Eaton DA (2014) PyRAD: assembly of de novo RADseq loci for phylogenetic analyses. Bioinformatics, 30, 1844-1849.
- Eaton DAR, Overcast I (2016) ipyrad: interactive assembly and analysis of RADseq data sets. Available at: http://ipyrad.readthedocs.io/.
- Drummond AJ, Suchard MA, Xie D, Rambaut A (2012) Bayesian phylogenetics with BEAUti and the BEAST 1.7. Molecular Biology and Evolution, 29, 1969-1973.
- Felsenstein J (2004) Inferring phylogenies. Sunderland, MA: Sinauer Associates.
- Gronau I, Hubisz MJ, Gulko B, Danko CG, Siepel A (2011) Bayesian inference of ancient human demography from individual genome sequences. Nature Genetics, 43, 1031-1034.
- Heled J, Drummond AJ (2010) Bayesian inference of species trees from multilocus data. Molecular Biology and Evolution, 27, 570–580.
- Lanfear R, Calcott B, Ho SYW, Guindon S (2012) PartitionFinder: combined selection of partitioning schemes and substitution models for phylogenetic analyses. Molecular Biology and Evolution, 29,1695-1701.
- Lanfear R, Frandsen PB, Wright AM, Senfeld T, Calcott B (2016) PartitionFinder 2: new methods for selecting partitioned models of evolution for molecular and morphological phylogenetic analyses. Molecular Biology and Evolution.
- Lemmon AR, Lemmon E (2008) A likelihood framework for estimating phylogeographic history on a continuous landscape. Systematic Biology, 57, 544–561.
- Peterson BK, Weber JN, Kay EH, Fisher HS, Hoekstra HE (2012) Double digest RADseq: an inexpensive method for de novo SNP discovery and genotyping in model and non-model species. PLoS One, 7, e37135.
- Raj A, Stephens M, and Pritchard JK (2014) fastSTRUCTURE: Variational Inference of Population Structure in Large SNP Data Sets. Genetics, 197, 573-589.
- Ronquist F, Teslenko M, van der Mark P, Ayres D, Darling A, et al. (2012) MrBayes v. 3.2: efficient Bayesian

phylogenetic inference and model choice across a large model space. Systematic Biology, 61, 539-542.

- Stamatakis A (2014) RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. Bioinformatics, 30, 1312-1313.
- Xie W, Lewis PO, Fan Y, Kuo L, Chen MH (2011) Improving marginal likelihood estimation for Bayesian phylogenetic model selection. Systematic Biology, 60, 150-160.

RECOMMENDED READING

- Unix shell background info here, here, here, and here.
- GNU Bash Reference Manual

TODO

- **Improve option and usage sections of `BEASTRunner.sh` script. DONE!** ✅
- **Add `MrBayes` scripts DONE!** ✅
- **Add options to `MrBayesPostProc` script, e.g. for burnin frac and stepping-stone MLE analysis. DONE!** ✅
- ** Give supercomputer scripts options (header w/flags) that will work for both a) TORQUE/PBS and b) SLURM Workload Manager cluster management and job scheduling systems (need meticulous work on this in `Super-pyRAD2PartitionFinder.sh`, `BEASTRunner.sh`, `BEASTPostProc.sh`, and `RAxMLRunner.sh`) **
- Make `pyrad` and `ipyrad` batch run scripts available
- Consider separate scripts to work with ipyrad
- Add capacity of adding or not adding path sampling/stepping-stone sampling to `BEAST` runs (use `BEASTRunner.sh` as springboard to develop tools in BEAST_PathSampling dir)

September 13, 2017 Justin C. Bagley, Richmond, VA, USA