Phylo* File Processing and Automation Using PIrANHA

Quick Guide for the Impatient

PIrANHA provides a set of tools for automating file processing and analysis in (phylo*=) phylogenetics, phylogenomics, and phylogeography.

> *NOTE:* By convention, in all content we refer to this software package as "PIrANHA" and we write the name of the main script in the package as `piranha` .

Downloading PIrANHA is easy, with two options: local install or Homebrew install.

Local install

```
## 1. Download latest release using git and move into piranha directory:
cd ~ ;
git clone https://github.com/justincbagley/piranha.git ./piranha-master/ ;
cd piranha-master/ ;

## 2. Install:
cd install/ ;
chmod u+x ./* ;
local_piranha ;
#
cp local_piranha ~/bin/ ; # makes installer available from command line, for future use (assuming ~/bin/ is in $PATH, as usual
```

Homebrew install

```
## 1A. Homebrew install 'by-hand':
brew tap justincbagley/homebrew-piranha
brew update
brew install piranha
echo "#################### HOMEBREW PIrANHA ALIAS:" >> ~/.bash_profile ;
echo "alias piranha=\"/usr/local/Cellar/piranha/0.4a2/piranha\"" >> ~/.bash_profile ; ## Add to ~/.bash_profile etc. for autor
source ~/.bash_profile ;

## OR 1B. Automated Homebrew install using installer:
cd ~ ;
git clone https://github.com/justincbagley/piranha.git ./piranha-master/ ;
cd piranha-master/ ;
cd install/ ;
chmod u+x ./* ;
brew_piranha ;
#
cp brew_piranha ~/bin/ ; # makes installer available from command line, for future use (assuming ~/bin/ is in $PATH, as usual
```

Get the `piranha` help text and a list of functions available in PIrANHA like so:

```
piranha -h
piranha -f list
```

Get the help text for a particular function:

```
piranha -f <function> -h
```

Convert between DNA alignment formats like so:

```
# Convert FASTA to PHYLIP:

    piranha -f FASTA2PHYLIP -f 1 -i <inputFASTA> -k 1 -v 1     # Single PHYLIP file
    piranha -f FASTA2PHYLIP -f 2 -k 1 -v 1                     # Multiple PHYLIP files

# Convert FASTA to VCF:

    piranha -f FASTA2VCF -i <inputFASTA> -o <output>

# Convert Mega to PHYLIP:

    piranha -f Mega2PHYLIP -i <inputMega> -k 1                 # Single Mega file
    piranha -f Mega2PHYLIP -m 1 -k 1                           # Multiple Mega files
# Convert NEXUS to PHYLIP:

    piranha -f NEXUS2PHYLIP -i <inputNEXUS> -v 1              # Single NEXUS file

# Convert PHYLIP to FASTA:

    piranha -f PHYLIP2FASTA -i <inputPHYLIP> -k 1             # Single FASTA file
    piranha -f PHYLIP2FASTA -m 1 -k 1 -v 1                    # Multiple FASTA files

# Convert PHYLIP to Mega:

    piranha -f PHYLIP2Mega -i <inputPHYLIP> -k 1             # Single PHYLIP file
    piranha -f PHYLIP2Mega -m1 -k 1                          # Multiple PHYLIP files

# Convert PHYLIP to NEXUS:

    piranha -f PHYLIP2NEXUS -i <inputPHYLIP> -p <partitionsFile> -f NEX     # Single PHYLIP file
    piranha -f PHYLIP2NEXUS -m 1                                            # Multiple PHYLIP files
```

Concatenate DNA sequence alignments (e.g. genes) like so:

```
# Create <taxonNamesSpaces> file with getTaxonNames function (creates file '<numTips>_taxon_names_spaces.txt'):

    piranha -f getTaxonNames -n <numTips>

# Concatenate PHYLIP alignments (e.g. 1 per gene):

    piranha -f concatenateSeqs -t <numTips>_taxon_names_spaces.txt

# Complete (fill in missing individuals) and concatenate PHYLIP alignments:

    piranha -f completeConcatSeqs -t <numTips>_taxon_names_spaces.txt
```

Phase consensus sequences from HTS (e.g. targeted sequence capture) using reference:

```
# Phase alleles with default settings (creates intermediate files and final, unaligned phased FASTAs):

    piranha -f phaseAlleles -i <input> -o <output> -r <reference>

# Phase alleles while masking reference indels (insertions/deletions) in final, unaligned phased FASTAs:

    piranha -f phaseAlleles -i <input> -o <output> -r <reference> -m 1
```

Run standard evolutionary analysis programs (run with -h for help text first):

```
# Run BEAST:

    piranha -f BEASTRunner

# Run ∂a∂i:

    piranha -f dadiRunner
    piranha -f dadiUncertainty

# Run RAxML:

    piranha -f MAGNET
    piranha -f RAxMLRunner

# Run RogueNaRok:

    piranha -f RogueNaRokRunner

# Run SNAPP:

    piranha -f SNAPPRunner
```

Conduct post-processing of results from standard evolutionary analysis programs (run with -h for help text first):

```
# Process output from BEAST:

    piranha -f MLEResultsProc
    piranha -f BEASTPostProc

# Process output from ExaBayes:

    piranha -f ExaBayesPostProc

# Process output from MrBayes:

    piranha -f MrBayesPostProc
```

Use the documentation links in the sidebar at *right* to navigate this documentation and learn more about PIrANHA, and contact the author for technical support or raise an issue.

PIrANHA Publications

We are working on a paper describing PIrANHA while we develop towards major release v1.0 (hopefully later this year, in 2020). However, the alpha pre-release versions of PIrANHA have been used in several of our publications, including:

- Bagley, J.C., Hickerson, M.J. and Johnson, J.B., 2018. Testing hypotheses of diversification in Panamanian frogs and freshwater fishes using hierarchical approximate Bayesian computation with model averaging. Diversity, 10(4), 120.
- Bagley, J.C., Mayden, R.L. and Harris, P.M., 2018. Phylogeny and divergence times of suckers (Cypriniformes: Catostomidae) inferred from Bayesian total-evidence analyses of molecules, morphology, and fossils. PeerJ, 6, p.e5168.
- Bagley, J.C., Uribe-Convers, S., Carlsen, M., Muchhala, N., 2020. Utility of targeted sequence capture for phylogenomics in rapid, recent angiosperm radiations: Neotropical *Burmeistera* bellflowers as a case study. Molecular Phylogenetics and Evolution. Available online. PubMed: https://www.ncbi.nlm.nih.gov/pubmed/32081762. doi: https://doi:10.1016/j.ympev.2020.106769.

Contact Info

Questions? Comments? Concerns? Bug fix requests? Feature requests for existing functions, or requests for new functions? If so, please get in touch here.

License

All code within **PIrANHA v0.4a2** repository is available "AS IS" under a 3-Clause BSD license. See the LICENSE file for more information.

Citation

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

- Bagley, J.C. 2020. PIrANHA v0.4a2. GitHub repository, Available at: http://github.com/justincbagley/PIrANHA.

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

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

DOI

The DOI for PIrANHA, via Zenodo (also indexed by OpenAIRE), is as follows: `DOI 10.5281/zenodo.596766`. Here are some examples of citing PIrANHA using the DOI:

Bagley, J.C. 2020. PIrANHA v0.4a2. GitHub package, Available at: http://doi.org/10.5281/zenodo.596766.

Bagley, J.C. 2020. PIrANHA. Zenodo, Available at: http://doi.org/10.5281/zenodo.596766.

April 17, 2020 Justin C. Bagley, St. Louis, MO, USA