Developing Reasonable Starting Trees with treeStartR

April M. Wright
Email april.wright@selu.edu
Southeastern Louisiana University
Biology Department
2400 N. Oak St
Hammond, LA. 70402

1 Summary

- Phylogenetic trees occupy a central role in comparative and taxonomic biology. Modern methods for inferring phylogenetic trees incorporate multiple data sources, and complex models.
- Including morphological taxa, and taxa that are known based on fossils but lack a morphological character matrix, can greatly increase the amount of missing data present in an analysis. In these situations, it can be challenging to find a starting tree with a computable likelihood for a Bayesian MCMC.
- treeStartR is an R package for efficiently adding taxa to an existing phylogeny. Using treeStartR, tips may be added at random, via the Most Recent Common Ancestor (MRCA) with other tips on the tree, or by user-defined taxonomy.
- treeStartR will allow for users to develop reasonable starting trees for phylogenetic analysis based on both data matrices and taxonomic expertise.

2 Keywords

Phylogeny, fossils, evolution, paleontology, R

3 Introduction

Phylogenetic trees occupy a central role in evolutionary biology, forming the backbone of much taxonomic (Sneath and Sokal 1973) and comparative work (Felsenstein 1985). The first phylogenetic trees were inferred using morphological data matrices. As genetic data became readily and abundantly available, methods development increasingly focused on maximum likelihood and Bayesian methods for estimating phylogeny from these data. Evidence that integration of the fossil record with molecular phylogenies improves comparative inference has lead to increased attention paid to methods for working with morphological data. Methods for working with combined datasets, sometimes termed "joint" or "total-evidence" methods are becoming increasingly popular, and have been extended to include "stratigraphic tips", tips known from fossil evidence, but which do not have character information. These stratigraphic tips are still useful for establishing clade ages on a dated phylogeny, but must be placed via expert analysis, rather than character information.

Phylogenetic trees, under all optimality criteria, estimate a startting tree, an initial tree that is improved throughout the analysis. Frequently used criteria for generating the starting tree include random addition of taxa (Stamatakis 2014; Bouckaert et al. 2014), parsimony (Stamatakis 2014), and clustering via neighbor-joining or UPGMA (Bouckaert et al. 2014). Another approach is drawing a tree from a tree prior (Höhna et al. 2014; 2017). However, all of these approaches can be challenging in the presence of stratigraphic taxa. Phylogenetic methods assume a common tree, i.e. a representation of the phylogeny on which all taxa in the analysis are present as tips, including stratigraphic taxa. Because stratigraphic taxa are placed via taxonomy, and not data, they cannot be added via typical phylogenetic algorithms. In practice, they are often added to the tree through the use of clade constraints, and are represented in the phylogenetic matrix as missing data (Höhna et al. 2014; 2017). No software presently simulates trees consistent with clade constraints, but instead reject starting trees inconsistent with the constraints. This is referred to as rejection sampling.

In an analysis assuming a model, an analysis with stratigraphic tips must not only find a starting tree, but

one with a computable probability given the model and the data. Practically, the addition of stratigraphic tips greatly increases the amount of missing data in an analysis because these taxa do not have character data. The net result of this is that a tree consistent with the constraints must be found with a reasonable likelihood in the presence of large amounts of missing data. This may be particularly challenging for parameter-rich models, which many models for incorporating stratigraphic taxa are (Heath et al. 2014).

Most phylogenetic software packages in wide use allow researchers to add a starting tree of their own, sidestepping the challenges of generating one. Commonly, a starting tree may be a tree from a prior study. While, under an appropriately-specified model, any initial conditions of a Bayesian MCMC will lead to convergence, a reasonable starting tree can speed this up. Hence, I introduce the package treeStartR to assist in generating reasonable starting trees for combined datasets, including stratigraphic taxa. This package includes a number of functions to add tips to a tree through a variety of methodologies, which can then be exported for use with most phylogenetic software. In the coming sections, I describe the usage of this package.

4 Description

treeStartR is written entirely in the R scientific computing language, and uses functionality from phytools (Revell 2012), ape (Paradis et al. 2004) and dplyr (Wickham et al. 2018). The core functionality of treeStartR is to take a total set of taxa, establish which are already on the phylogeny, and add those that are not. The taxa that are not on the phylogeny can be added in three principal ways:

4.1 Functions

- present_tippr: This approach looks at the genus of each taxon not represented on the tree. If there is another member of that genus, or multiple members of that genus, on the tree, they will guide the addition of the taxon. If one member of the genus is on the tree, the new taxon will be added subtending the parent node of its congener. If multiple members of the same genus are present, the taxon will be added subtending the most recent common ancestor (MRCA) of its congeners. This function is effective for efficiently adding taxa that are closely-related to other taxa on the tree.

- rand_absent_tippr - This approach draws a random node from the phylogeny for each taxon not represented on the tree. The taxon is then added subtending this random node. This function is designed to create starting trees if the relationships of the taxa to be added are truly unknown, or to generate null starting trees for examining convergence efficiency.

- absent_tippr - This method offers a user interface. In this method, a visualization of the tree with node labels will pop up. The user then decides where they would like the taxon to be added to the tree, and enters the node number they would like it to subtend. The tree visualization then updates, as the node labels change as tips are added to the tree. The researcher will specify node numbers until all taxa are added to the tree. This approach is useful for small amounts of taxa to be added to a tree.

- text_placr - It would be inconvenient and unweildy to place all taxa on a large tree via user interface. Therefore, the text_placr function allows users to create an input file with the names of the tips to be added, where the tips will be added. In this function, the datafile is read in. For each tip, a taxon or list of taxa that represent where the tip should be placed is read in. If there is one taxon that represents where the tip should be placed, the tip is added subtending the parent of that taxon. If there are multiple, the tip will be added subtending the MRCA of all of them. This function has been made available to facilitate taxonomically-guided bulk addition of tips.

4.2 Test Suite

treeStartR has a test suite implemented via testthat. The software is automatically tested at each change using Travis CI. The test passage status and coverage percentage can be viewed on the software's website (https://wrightaprilm.github.io/treeStartR/). Examples of the input files can be found in the test suite.

5 Example Analyses

The treeStartR package comes with a test dataset, bears. These data come from Kraus and Abella, and were previously compiled for use with fossilized birth-death dating by Heath et al. A starting tree is provided, and seven taxa are in the total set of tips that are not present on the tree. Of these seven tips, two have congeners and can be placed on the tree via the present_tippr function. Five do not, and must be placed via one of the other functions. Examples of how to make a complete tree follow.

For all below analyses, first install and load treeStartR, and load the included dataset.

```
install.github("wwrightaprilm/treestartr")
library(treestartr)
data(bears)
```

This dataset contains all the input files needed to add taxa to trees. I will note, in the instructions, where helper functions are used to generate these files for your own data.

5.1 Automatically placing taxa via taxonomy

The present_tippr() function allows researchers to place a tip on the tree by locating other species of the same genus. As input, this function takes a phylogenetic tree (a phylo object, which can be read in via ape) and a list of taxa that are not currently represented on the tree. This list is provided with the data package included with treestartr, but can be generated through the use of user functions dataf_parsr() and genera_strippr() from a CSV or TSV file containing two columns (taxon and age). present_tippr calls two utility functions, make_tree_df() and get_found(), which make a list of genera found on the tree and match the genera of the taxa in the absent list to those genera, respectively. If the genus of a taxon in the absent_list has multiple congeners on the tree, present_tippr calls find_MRCA from ape to find the most recent common ancestor of all matching taxa, and places the new tip subtending this node. If the genus matches one congener, present_tippr uses the getParent() function from ape to get the parent node. The new tip is then added subtending this node. For compatability with the pipe function, the return type of this function is a phylo object, and the input tree is the first argument to the function:

```
new_tree <- present_tippr(tree, absent_list)
```

The output phylogenetic tree will have all the taxa present on the original starting tree, as well as the tips that could be identified with a genus on the tree. In the case of the test data, the initial tree has 15 tips, and the output tree has 17 tips. The tips added by this function are Indarctos_punjabiensis and Ursus_abstrusus.

Taxon	Clade
Kretzoiarctos_beatrix	*Indarctos_arctoides*
Kretzoiarctos_beatrix	*Indarctos_vireti*
Ursus_abstrusus	*Ursus_arctos*
Ursus_abstrusus	*Ursus_spelaeus*
Ursus_abstrusus	*Ursus_americanus*

Table 1: Example data table input for function text_placr

5.2 Manually placing taxa via taxonomy

Tips can also be placed manually. This function plots the input phylogeny, with node labels. The researcher chooses the number of the node they would like the new tip to subtend. If called on the included example tree, like so:

```
new_tree <- absent_tippr(tree, absent_list)
```

5 tips will be added for a total of 20 tips on the tree. The return type of this function is a phylogenetic tree, containing all the tips of the input tree and all of the tips that were added to it.

5.3 Placing taxa via taxonomy datafile

Adding tips manually is not practical if there are many tips to be added. In this case, an input CSV file can be provided. One is included with the example data, but this file can be created in any spreadsheet editor. In the input file format, one column will be the tip to be added. The other column is the group to which it should be added. If there is only one taxon on the tree that will be used for placement, the taxon to be added will have a single row. In this case, the taxon will be added via getParent(). If there are multiple tips, the taxon to be added will be represented in multiple rows. Each row will have the same taxon represented, but a different taxon (from the tree) listed in the second column. An example is below:

In this case, tip *Kretzoiarctos_beatrix* will be added subtending the MRCA of *Indarctos_arctoides* and *Indarctos_vireti*. *Ursus_abstrusus* would be added subtending the MRCA of *Ursus_arctos*, *Ursus_spelaeus*, and *Ursus_americanus*. Called on the example data:

```
new_tree <- text_placr(tree, mrca_df)
```

the returned tree will contain 17 tips.

5.4 Automatically placing taxa on the tree at random

The final main function of treeStartR is rand_absent_tippr. It accepts as input a tree and a list of taxa to be added to it:

```
new_tree <- rand_absent_tippr(tree, absent_list)
```

For each tip to be added, a node is selected on the tree at random with the sample() function. The tip to be added is then added subtending this node. Called on the example tree, the returned new_tree will have 20 tips.

6 Significance

The bear data set is an excellent test dataset, because it is small and well-behaved. However, the real utility in using a starting tree in a phylogenetic analysis becomes more apparent when the phylogenetic problem is difficult - that is, when missing data, parameter richness, and/or taxon richness cause finding starting parameters to be challenging. To demonstrate the utility of treeStartR for these large problems, I've also included a second dataset from the Formicidae.

This phylogeny contains 666 taxa with molecular data from Blanchard and Moreau 2017. There are an additional 91 taxa known from the fossil record. I added the unknown tips to the phylogeny using a combination of functions in tresStartR.

First, I loaded in the ant data.

```
data(ants)
```

Then, we verify that the number of taxa in the total dataset, but not on the tree is 91:

```
absent_list <- genera_strippr(ant_tree, tax_list)
```

I first place the taxa that have congeners on the tree. Then, I regenerate the list of taxa that are in the total analysism but not on the tree. There should be 21 taxa remaining.

```
new_tree <- present_tippr(ant_tree, absent_list)
absent_list <- genera_strippr(new_tree, tax_list)
```

The data package contains a CSV file with information on phylogenetic placement for 16 of these taxa. I then use the textplacR function to place these.

```
genus_tree <- text_placr(new_tree, mrca_df)
absent_list <- genera_strippr(genus_tree, tax_list)
```

The final five taxa can be placed either at random or by taxonomic knowledge.

Finally, any polytomies in the tree are broken up, and the tree is exported.

```
bifurcating_tree <- multi2di(genus_tree)
write.nexus(bifurcating_tree, file ="starting_tree.tre")
```

In the Bayesian phylogeny estimation program RevBayes (**?**), I began a Bayesian combined molecular-morphological phylogenetic estimation. I initialized my analysis both without a starting tree (tree drawn from a uniform distribution of trees), and with. I attempted to initialize the analysis 3 times for each treatment. In the analysis without a starting tree, an initial set of parameter values for the Bayesian MCMC was not found in RevBayes' set number of attempts (100) to locate them. Without these values, the MCMC does not begin, and a phylogenetic tree is not computed. With a starting tree, initial values were found within 8, 17, and 10 attempts. This demonstrates on a large empirical dataset that inputting from a reasonable starting tree can improve the efficiency of analysis. Code to reproduce this test is on <dryad whatever>.

7 Citation information

To be added prior to publication.

8 Data Accessibility

All data are packaged with the software, and will be downloaded during the software installation via GitHub.

References

Blanchard, B. D. and C. S. Moreau. 2017. Defensive traits exhibit an evolutionary trade-off and drive diversification in ants. Evolution 71:315–328.

Bouckaert, R., J. Heled, D. Kühnert, T. Vaughan, C.-H. Wu, D. Xie, M. A. Suchard, A. Rambaut, and A. J. Drummond. 2014. BEAST 2: a software platform for Bayesian evolutionary analysis. PLoS computational biology 10:e1003537.

Felsenstein, J. 1985. Phylogenies and the comparative method. The American Naturalist Pages 1–15.

Heath, T. A., J. P. Huelsenbeck, and T. Stadler. 2014. The fossilized birth-death process for coherent calibration of divergence-time estimates. Proceedings of the National Academy of Sciences 111:E2957–E2966.

Höhna, S., T. A. Heath, B. Boussau, M. J. Landis, F. Ronquist, and J. P. Huelsenbeck. 2014. Probabilistic graphical model representation in phylogenetics. Systematic Biology 63:753–771.

Höhna, S., M. J. Landis, and T. A. Heath. 2017. Phylogenetic inference using RevBayes. Current Protocols in Bioinformatics .

Paradis, E., J. Claude, and K. Strimmer. 2004. Ape: analyses of phylogenetics and evolution in r language. Bioinformatics 20:289–290.

Revell, L. J. 2012. phytools: an r package for phylogenetic comparative biology (and other things). Methods in Ecology and Evolution 3:217–223.

Sneath, P. and R. Sokal. 1973. Numerical taxonomy. Springer.

Stamatakis, A. 2014. Raxml version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. Bioinformatics 30:1312–1313.

Wickham, H., R. FranÃğois, L. Henry, and K. MÃijller. 2018. dplyr: A Grammar of Data Manipulation. R package version 0.7.6.

Version dated: July 18, 2018