Bioinformatics, 34(6), 2018, 1053–1055
doi: 10.1093/bioinformatics/btx701
Advance Access Publication Date: 30 October 2017
Applications Note

Phylogenetics

Efficient comparative phylogenetics on large trees

Stilianos Louca[1,2,*] **and Michael Doebeli**[1,2,3]

[1]Biodiversity Research Centre, University of British Columbia, Vancouver, BC, V6T1Z4, Canada, [2]Department of Zoology, University of British Columbia, Vancouver, BC, V6T1Z4, Canada and [3]Department of Mathematics, University of British Columbia, Vancouver, BC, V6T1Z4, Canada

*To whom correspondence should be addressed.

Associate Editor: Alfonso Valencia

Received on June 8, 2017; revised on September 27, 2017; editorial decision on October 26, 2017; accepted on October 28, 2017

Abstract

Motivation: Biodiversity databases now comprise hundreds of thousands of sequences and trait records. For example, the Open Tree of Life includes over 1 491 000 metazoan and over 300 000 bacterial taxa. These data provide unique opportunities for analysis of phylogenetic trait distribution and reconstruction of ancestral biodiversity. However, existing tools for comparative phylogenetics scale poorly to such large trees, to the point of being almost unusable.
Results: Here we present a new R package, named ‘castor’, for comparative phylogenetics on large trees comprising millions of tips. On large trees castor is often 100–1000 times faster than existing tools.
Availability and implementation: The castor source code, compiled binaries, documentation and usage examples are freely available at the Comprehensive R Archive Network (CRAN).
Contact: louca.research@gmail.com
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The advance of high-throughput sequencing generates whole genome sequences and marker gene-based phylogenies at rapidly increasing rates. For example, the SILVA 16S ribosomal RNA reference tree currently contains ∼570 000 bacterial and archaeal tips (Quast *et al.*, 2013). Further, machine learning algorithms enable the automated inference of metabolic phenotypes for thousands of microbial genomes (Karp *et al.*, 2010). These phylogenetic and phenotypic data provide unprecedented opportunities for large-scale evolutionary analysis, such as reconstruction of past metabolic diversity and phenotype predictions for poorly characterized extant clades. However, these vast data also present a serious challenge for existing phylogenetics tools. Most existing phylogenetic packages (e.g. FitzJohn, 2012; Paradis *et al.*, 2004; Revell, 2012) have been designed for much smaller trees containing at most a few thousand tips, and thus scale poorly to large datasets. For example, a simple task such as pruning tips from the SILVA tree can take several hours on a modern laptop using the popular software package `ape` (Paradis *et al.*, 2004). Similarly, ancestral state reconstruction (ASR) for a binary trait with standard continuous-time Markov models (Mk models) takes several hours on the SILVA tree using `ape`. A simple power-law analysis reveals that these functions exhibit a time complexity that scales roughly quadratically with tree size. Re-designed efficient algorithms for large-scale phylogenetic analysis are thus urgently needed. This need is intensified when reconstructions are nested into cross-validation or bootstrapping algorithms. As we explain below, super-linear time complexities can be avoided using redesigned algorithms optimized for large trees.

Here we present a new package for the R statistics environment that enables phylogenetic analysis using algorithms optimized for large trees. We named this package ‘`castor`’, after the animal able to fell large trees. `castor` emerged as part of our work on large microbial phylogenies (including hundreds of thousands of strains), which necessitate more efficient implementations of common functions than currently available.

2 Materials and methods

2.1 `castor`: a collection of highly optimized algorithms

`castor` provides efficient implementations of common phylogenetic functions, focusing on analysis of trait evolution on fixed trees.

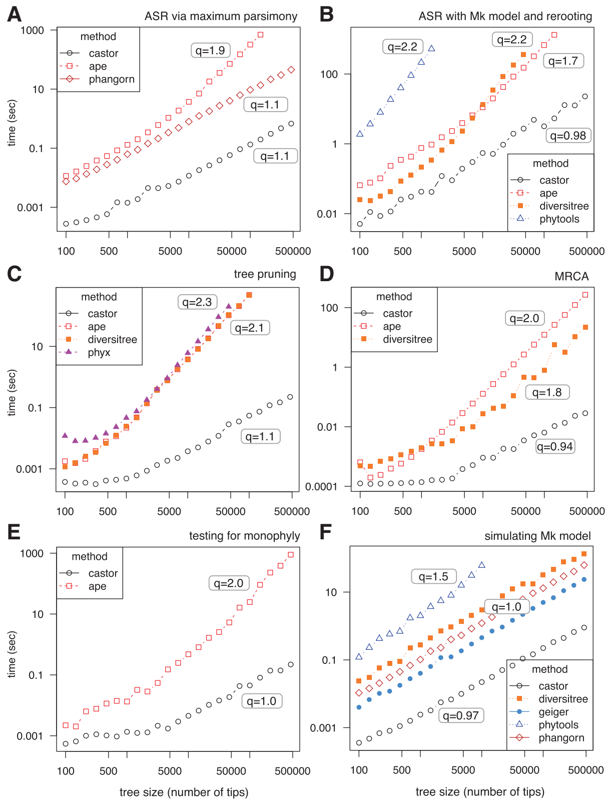

Fig. 1. Comparison of computation time needed for various tree operations in castor and other packages (time T over tree size S). (**A**) ASR of a discrete trait with 5 states, using maximum-parsimony. (**B**) ASR of a discrete trait with 5 states, using an 'equal rates' Mk model with rerooting. (**C**) Pruning trees by removing half of the tips. (**D**) Determining the most recent common ancestor (MRCA) of two random tips. (**E**) Testing whether a subset of tips is monophyletic. (**F**) Simulating an Mk model (5 states) for discrete trait evolution. Note the logarithmic axes in all figures. Package names are listed in the legends. Fitted power-law exponents ($T \propto S^q$) are shown next to every curve. Compared packages include phyx (Brown *et al.*, 2017), ape (Paradis *et al.*, 2004), diversitree (FitzJohn, 2012), phytools (Revell, 2012), phangorn (Schliep, 2011) and geiger (Harmon *et al.*, 2008). Detailed functions and options used are explained in Supplementary Material S2. For additional benchmarks of other functions see Supplementary Figure S1

Notably, castor provides functions for ASR of discrete traits, for example using Mk models (Yang *et al.*, 1995) or maximum-parsimony methods, as well as ASR of continuous traits, for example using squared-change parsimony (Maddison, 1991). Further, castor includes functions for hidden state prediction, i.e. for estimating a priori unknown trait values (states) in tips based on a subset of tips with known states (Zaneveld and Thurber, 2014). castor also enables statistical analysis of trait distribution, such as calculating phylogenetic autocorrelation, for simulating or fitting models of trait evolution, as well as for common operations such as tree pruning, inference of most recent common ancestors or calculating distances between tips. castor fully supports monofurcating and multifurcating trees, in contrast to the majority of existing tools that generally require bifurcating trees.

2.2 Comparison to existing software

Most of castor's functions exhibit a time complexity that scales linearly with tree size, in many cases achieving a 100- to 1000-fold efficiency when compared to existing packages (Fig. 1 and Supplementary Fig. S1). For example, removing 50% of the tips from the SILVA tree takes less than 1 s on a modern laptop using castor and over 4 h using the package ape (Fig. 1C). Similarly, maximum-likelihood ASR of a discrete trait on the SILVA tree using an Mk model (5 possible states, all rates equal) takes about 30 s using castor and over 4 h using ape (Fig. 1B). castor's high efficiency is achieved in multiple ways. First, dynamic programming algorithms are used wherever possible. Second, most algorithms benefit from auxiliary data structures that are temporarily created on demand. For example, calculation of most recent common ancestors is achieved in linear time (Fig. 1D) by using a lookup table that maps each node to its parent node, instead of repeatedly searching for each node's parent amongst all possible nodes. Third, in certain ASR algorithms involving rerooting (e.g. maximum-likelihood Mk models) redundant calculations are avoided by storing previously computed intermediate quantities (see Supplementary Material S1). Fourth, ASR of discrete traits using Mk models, which requires repeated exponentiation of the Markov transition matrix along each edge, was accelerated through an ad-hoc exponentiation algorithm that becomes highly efficient when the same matrix is exponentiated several times. Fifth, castor is almost entirely implemented in C++, a programming language optimized for high-performance computations.

3 Conclusion

castor is a collection of highly efficient algorithms for phylogenetic analysis on large trees, easily scaling to millions of tips. Although castor focuses on analyzing trait distributions and reconstructing trait evolution, it also includes several other common functions for working with phylogenies. On large trees, castor performs many of these functions orders of magnitude faster than other comparable packages, thereby enabling large-scale phylogenetics using substantially reduced computational resources.

Acknowledgements

We thank Laura W. Parfrey and Matthew Pennell for helpful discussions and for providing comments on our manuscript.

Funding

S.L. and M.D. were supported by the Natural Sciences and Engineering Research Council of Canada (NSERC). SL was also funded by the Biodiversity Research Centre, UBC. The authors declare no conflict of interest.

Conflict of Interest: none declared.

References

Brown,J.W. *et al.* (2017) Phyx: phylogenetic tools for unix. *Bioinformatics*, **33**, 1886–1888.

FitzJohn,R.G. (2012) Diversitree: comparative phylogenetic analyses of diversification in r. *Methods Ecol. Evol.*, **3**, 1084–1092.

Harmon,L.J. *et al.* (2008) GEIGER: investigating evolutionary radiations. *Bioinformatics*, **24**, 129.

Karp,P.D. *et al.* (2010) Pathway tools version 13.0: integrated software for pathway/genome informatics and systems biology. *Brief. Bioinform.*, **11**, 40–79.

Maddison,W.P. (1991) Squared-change parsimony reconstructions of ancestral states for continuous-valued characters on a phylogenetic tree. *Syst. Biol.*, **40**, 304–314.

Paradis,E. *et al.* (2004) APE: analyses of phylogenetics and evolution in R language. *Bioinformatics*, **20**, 289–290.

Quast,C. *et al.* (2013) The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. *Nucleic Acids Res.*, **41**, D590–D596.

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

Schliep,K.P. (2011) phangorn: phylogenetic analysis in r. *Bioinformatics*, **27**, 592–593.

Yang,Z. *et al.* (1995) A new method of inference of ancestral nucleotide and amino acid sequences. *Genetics*, **141**, 1641–1650.

Zaneveld,J.R.R. and Thurber,R.L.V. (2014) Hidden state prediction: a modification of classic ancestral state reconstruction algorithms helps unravel complex symbioses. *Front. Microbiol.*, **5**, 431.