Advanced Molecular Evolution using *HyPhy*

Sergei L. K. Pond, Art F. Y. Poon, and Spencer V. Muse

1 Introduction

1.1 What is *HyPhy*?

***HyPhy* is a software package.** Its name stands for '**Hy**pothesis testing using **Phy**logenies'. So, the primary function of *HyPhy* is to analyze genetic (DNA or RNA) sequences in a phylogenetic framework. A phylogeny is a hypothesis — it is a tree-shaped (hierarchical) model of how sequences are descended from common ancestors. Because of these evolutionary relationships, sequences are not independent random outcomes of molecular evolution. Put another way, one should not get too excited about observing hundreds of sequences that all contain an 'A' at position 101 because they are very probably all copies of an older sequence that contained an 'A' at position 101. If you want to test *any* hypothesis that involves sequence data, then you have to take the phylogeny into account. Because of this, phylogenies are a core component of *HyPhy*.

***HyPhy* is scientific software.** It was initially developed by Sergei Pond to clean up and extend some programs that were written by Spencer Muse. Because *HyPhy* was designed to be highly customizable, it became useful to simply keep extending *HyPhy* as new research projects arose. *HyPhy* became an open-source project under the General Public License (GPL), allowing anyone to download, use and modify the source code. It is also distributed for free as executable binaries that are compiled for Mac OS X, Windows, and Linux. At the time of writing this book, *HyPhy* has over 5000 registered users and has been cited in over 500 peer-reviewed scientific publications.

***HyPhy* is as simple or as complex as you need it to be.** It has a rich graphical user interface that can display colourful tables, charts, sequence alignments and trees; to design an analysis using a point-and-click interface; and to choose from a wide selection of packaged analyses in a standard menu. On the other hand, *HyPhy* also has its own scripting language for implementing as complex an analysis as you can imagine. It can be compiled as a shared library and loaded into another programming environment such as Python, so that it can act as a component of a bioinformatic pipeline. It can be run in a parallel computing environment where an analysis is broken down into tasks that can be distributed among tens or hundreds of processors and run simultaneously.

For the purpose of this book, *HyPhy* is **a scripting language for modeling molecular evolution**. The fundamental objects of *HyPhy* not only include integers, matrices and associative lists, but also sequence alignments, trees, and substitution rate models.

1.2 Is this book for me?

This book was written for you. You are a graduate student or post-doc who once mentioned that you wrote a program once and, as a result, you've been asked by your lab principal investigator[1] to run analysis $\mathcal{X}$ on some data. $\mathcal{X}$ is similar to $\mathcal{Y}$, a standard analysis that has been used hundreds of times by other laboratories and is available in dozens of software packages and web applications.

But, owing to the uniqueness of the model system being studied in the lab, or the research interests of the PI, there is a significant difference that means that you can't run $\mathcal{X}$ using software designed to run $\mathcal{Y}$. Or any software that you can find. Just as you've resigned yourself to writing your own #$!&@ program to implement $\mathcal{X}$, one of your colleagues mentions recently seeing a similar analysis in a paper. After downloading and scanning over the paper, you discover two things: (1) the analysis is definitely not $\mathcal{X}$, but (2) it was done in something called *HyPhy*.

HyPhy enables you to do *anything* in the domain of phylogenetics. This is a problem — to borrow a popular phrase from C programming, it "gives you enough rope to hang yourself with". While there is a large and growing number of biologists that use the standard analyses distributed with *HyPhy* or accessed as web applications on our public computing cluster (*Datamonkey*),

[1] We know that you're not the principal investigator, because no PI would have the time to be reading this right now.