IPS Instituto Politécnico de Setúbal
Escola Superior de Tecnologia do Barreiro

Instituto Politécnico de Setúbal

Escola Superior de Tecnologia do Barreiro

Laboratório em Bioinformática

Licenciatura em Bioinformática

Automagic phylogenies

January, 2023

Group
Duarte Valente (202000053)
Gonçalo Alves (202000170)
Matilde Machado (202000174)
Rodrigo Pinto (202000177)
Guilherme Silva(202000178)
Marine Fournier(202000224)

Contents

1 Introduction

This report provides an overview of a software program designed to generate phylogenetic trees. Phylogenetic trees are graphical representations of evolutionary relationships among species or groups of organisms. The program utilizes various algorithms and data inputs to generate accurate and comprehensive phylogenetic trees. This report will provide a brief overview of the features and capabilities of the software, as well as its intended use and target audience. The software program is designed to be user-friendly and accessible for both researchers and educators in the field of evolutionary biology. It integrates advanced algorithms for tree construction, allowing for the analysis of large and complex datasets. The program also includes visualization tools for tree presentation, as well as options for customizing and annotating the tree output. Additionally, the software can import and export data in a variety of formats, making it easy to integrate with other analysis tools. The program is intended to provide a comprehensive and efficient solution for phylogenetic tree construction and analysis, and is an essential tool for anyone studying evolutionary relationships among species or groups of organisms.

2 Background

In this section will be provided a quick background information on the field of phylogenetics and the challenges associated with creating phylogenetic trees.

Phylogenetics is a study that aims to understand the evolutionary relationships among vast groups of similar organisms. It uses molecular biology to achieve to compare the genetic and morphological characteristics of different organisms, infering their evolutionary relationships. The main goal of this process is to construct evolutionary/phylogenetic trees, which depict the evolutionary relationships among different organisms.

The process of creating a phylogenetic tree can be challenging. One of the main challenges is the availability of data. For example, it can be difficult to obtain high-quality genetic data for a certain group of organisms. The complexity of determining evolutionary relationships can be compounded by various factors such as, the method used, the type of data, and the assumptions made. Also, the construction of a phylogenetic tree assumes that similarities among organisms are the result of a shared ancestry, but it's possible that similarities may have developed independently within different groups of organisms.

Finally, creating a phylogenetic tree requires making choices about the appropriate model and the appropriate method for inferring relationships, like Maximum likelihood, Bayesian and Distance-based methods. Selecting the most fitting model can be challenging, but

fortunately, there are resources available to assist in making the best choice.
The field of phylogenetics requires expertise from multiple areas, like molecular biology and computer science. Phylogenetic trees can offer significant insights into the evolutionary connections among organisms, however, it is crucial to keep in mind the difficulties and ambiguities that can arise during the creation of these trees.

3 Methodology

The methodology for building phylogenetic trees involves several key steps, including data acquisition, data processing, and tree inference. Data acquisition involves obtaining high-quality genetic or morphological data for each of the organisms being analyzed. Data processing involves cleaning, organizing, and transforming the data into a format that is suitable for tree inference. Finally, tree inference involves using a variety of algorithms and models to construct the phylogenetic tree based on the processed data. In the next subsections it will be explained every part of the program from the data acquisition to the tree build and what methods were used.

3.1 Data Acquisition

For the data acquisition we used the EntrezAPI, which The Entrez API is a component of the NCBI (National Center for Biotechnology Information) programmatic access to the vast collections of data maintained by the NCBI. The API provides a set of programmatic tools for accessing NCBI databases, including PubMed, GenBank, and others. The API allows developers to retrieve and manipulate data in a format that is suitable for analysis and integration into other programs or applications. The API supports a wide range of programming languages and platforms, making it a versatile and convenient tool for a wide range of scientific, medical, and research applications. The API is designed to be flexible, allowing developers to specify the data they need and the format they want it in, while also providing a variety of options for filtering, sorting, and transforming data to meet their specific needs. Overall, the Entrez API provides a powerful and flexible tool for accessing NCBI data and integrating it into a wide range of scientific and medical applications.
The final output of the stage is a folder full of FASTA files that we will use to build the tree.

3.2 Data processing

As soon as the FASTA files were gathered, there are four necessary steps to make this FASTA files operable.

The initial procedure involves assigning new, more descriptive labels to each sequence contained within each FASTA file, thereby rendering the information contained within each sequence more readily comprehended.

The second step entails the alignment of each FASTA file, which will be executed utilizing the MAFFT software, which is a software tool for multiple sequence alignment. It can align large numbers of sequences efficiently and accurately, making it a popular choice in the field of molecular biology and genetics.

The third operation involves the meticulous concatenation of all FASTA files, culminating in the creation of a comprehensive and unified file. To make this possible we used only python dictionaries. This procedure paves the way for conducting a maximum likelihood tree analysis, yet to undertake a Bayesian tree analysis, a final step is require.

Finally, the final step involves tranforming the FASTA file into the NEXUS file format, with the aid of the SeqMagick tool, to optimize the data for further analysis, culminating in the creation of a Bayesian phylogenetic tree. SeqMagick is a powerful and flexible software package that allows for efficient manipulation and conversion of multiple sequence alignment (MSA) files in various formats.

3.3 Tree Inference

The end result of this phase will be the production of two distinct phylogenetic trees - one generated through the Maximum Likelihood method and the other, a Bayesian tree.

To construct the maximum likelihood tree, it was imperative to utilize a FASTA file and the software of choice was RaxML. RaxML is a popular open-source software that uses maximum likelihood algorithms to construct phylogenetic trees from molecular sequence data. The implementation of RaxML within this project allowed for the efficient and accurate creation of the maximum likelihood tree.

The final step in this phylogenetic tree building procedure will entail utilizing the NEXUS file and the MrBayes software to construct a Bayesian tree. A brief introduction to MrBayes, a widely-utilized software for Bayesian inference of phylogenetic trees, must also be provided to contextualize its application in this process.

3.4 Test Suite

To ensure the accuracy of our results, a comprehensive test suite was implemented throughout the phylogenetic tree building process. The test suite consisted of a series of validation checks for each step, starting the data acquisition with EntrezAPI, to the final phylogenetic tree output using MrBayes. The validation checks included verifying the integrity of input data, examining the alignment quality, and evaluating the consistency of tree topology. The results of the tests were recorded and analyzed to ensure that the program meets the required standards and produces accurate phylogenetic trees. The test were implment using pytest. Pytest is a popular testing framework for Python programming language that is used to write and run tests for applications and libraries. It allows developers to write tests in a simple, concise and readable manner. Pytest provides a rich set of tools for writing tests, including fixtures for setup and teardown, assertion introspection and plugins for extended functionality.

4 Implementation

The implementation phase is a crucial step in the development process where the methodology and techniques outlined in the previous stages are put into action. This phase involves executing the steps and methods that were designed and tested in the planning phase, resulting in the creation of a functional product or solution. The implementation phase, much like its predecessor, the methodology phase, will encompass four distinct stages of execution.

4.1 Data acquisition

4.2 Data Processing

After getting the FASTA folder, it was necessary to change the names of each sequence in each FASTA file inside the folder. The list of names was obtained from the FiltredScientificNames$_{l}ist.txt(anout$

4.3 Tree Inference

4.4 Test Suite

- Includes the specific programming languages, libraries, and tools used to develop the program. It also includes information about the programming techniques that were employed and any specific coding practices that were followed.

- This section is where the code itself is typically included or referenced, and it should be detailed enough for someone with a similar level of expertise to understand how the program works and could potentially make changes or modifications to the code. For this we used SeqIO from Bio library. SeqIO is a module in the Biopython library used for reading and writing biological sequence files, such as FASTA and GenBank files.

5 Results

- Colocar os mambinhos dos graficos e exemplis de fastas, alignments, concatenate, etc
- Summary of the results obtained from testing the software, including any performance metrics and examples of the generated phylogenetic trees.
- Observations or insights gained from the results, and how they compare to expected or previous results.
- It should provide any visualizations or plots that help to interpret the results and explain any patterns or trends found in the data.

6 Conclusion

References

[1] Risso D, Schwartz K, Sherlock G, Dudoit S (2011). GC-Content Normalization for RNA-Seq Data. *BMC Bioinformatics*, 12(1), 480

[2] asdfasdfasdf , 12(1), 480