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 Tree Inference

4 Implementation

- 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 Antigos Requirements

6.1 Operating System

The software must be compatible with Linux.

6.2 Software Specifications

- Docker v20.10.22 &
- Snakemake &
- Python 3.10.9
- Biopython 1.77 (seqmagick)
- Biopython 1.80 (18 November 2022)

- Mafft v7.490
- RaxML 8.0.0
- Modeltest-ng
- Mrbayes

6.3 Inputs and Outputs

As input the program will need 4 arguments:
1 - Scientific name of the species
2 - Taxonomy Hierarchy
3 - Proximity value (Proximity values would indicate how closely related two organisms are, the higher the percentage, higher the relationship between organisms)
4 - Simularity value (Similarity value is a measure of how alike two or more sequences or organisms are, based on their genetic or physical characteristics. The higher the similarity value, the more similar these two organisms are.)
In the end as output the program will generate 2 pdfs with phylogenetics trees.

7 Antigo Design

- Preencher
- Architecture: This includes a high-level overview of the overall structure and organization of the program, including any major components or modules and how they interact with each other.
- Algorithms: This includes a detailed explanation of any key algorithms or computational methods used in the program, including any trade-offs or decisions made in their selection.
- Data structures: This includes information about the specific data structures used to store and organize data within the program, and how they support the algorithms and overall program architecture.
- User interface: This includes information about how the program is intended to be used by the end-user, including any specific user interface elements (such as buttons, menus, etc.) and how they function.

8 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