Assignment-2

karthik.viswanathan

September 2020

1 Introduction

In this report, I will be discussing about the work done , methods used and assumptions made while working on this assignment based on creating phylogenetic trees using UPGMA method from scratch. There are four files in which the codes have been written are namely `s11.py` , `s12.py` , `s21.py` , `s22.py`.

2 Question 1

The files associated with question one are `s11.py` , `s12.py` , `nucleotide_alignment.txt`, `Ndistance.csv` and `a12.txt`. `s11.py` includes code to extract sequences from `nucleotide_alignment.txt` and calculate distance matrix taking into account the gaps. The distances have been calculated in the following manner:

$$dist(i)(j) = find\_hamming(seq\_i, seq\_j)/len(seq\_i - r1) \tag{1}$$

Here, $r1$ represents the number of instances when two gaps are present in the same index in $seq(i)$ and $seq(j)$. The results are stored in `Ndistance.csv` and the distance matrix obtained from it is used in `a12.txt` to calculate the UPGMA tree in newick format. The branch lengths for the tree is given using:

$$branch\_length = min(dist)/2 \tag{2}$$

Here, $min(dist)$ represents the minimum value in the distance matrix which are about to be clustered. The newick tree format is stored in `a12.txt`.

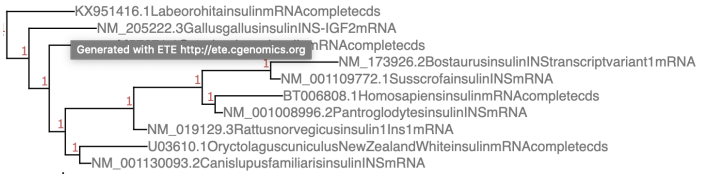

Figure 1: Phyl-1

3 Question 2

The files associated with question one are `s21.py` , `s22.py` , `protein_alignment.txt`, `Pdistance.csv`, `BLOSUM.txt` and `a22.txt`. `s21.py` includes code to extract sequences from `protein_alignment.txt` and calculate distance matrix taking into account the scores from `BLOSUM.txt` . The distances have been calculated in the following manner:

$$dist(i)(j) = find\_dist(seq\_i, seq\_j, score) \tag{3}$$

Here, the *score* is a dictionary involving a list of all atomic pairs and their scores/distances. The results are stored in `Pdistance.csv` and the distance matrix obtained from it is used in `a22.txt` to calculate the UPGMA tree in newick format. The branch lengths for the tree is given using:

$$branch\_length = 1000/score(a)(b) \tag{4}$$

Reciprocal has been taken as two sequences with maximum scores and clustered over others and hence the branch length between them is the least. To felicitate this convention, a reciprocal has been taken to avoid discrepancies.

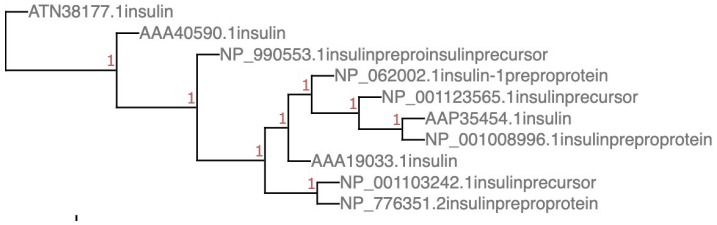

Figure 2: Phyl-2

4 Libraries used

No external libraries have been used other than Numpy. JSON and Pandas have been used to only convert and indent files so as to store them in text files.

5 Assumptions

While calculating the distance matrix in `q2` , it has bee assumed that if both the sequence have gap at a given index, it can be ignored and that gap opening starts for a given sequence if one sequence has a gap and other does not and that the sequence with the gap has a non-gap in its previous index / gap in its previous sequence given the latter sequence also has a gap.