Integrative Transcriptomics
Prof. K. Nieselt,
Institute for Bioinformatics and Medical Informatics Tübingen
Prof. S. Nahnsen,
Institute for Bioinformatics and Medical Informatics Tübingen

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Lecture: Grundlagen der Bioinformatik — SoSe 2023

Assignment 5 (20 points)

Hand out: Thursday, May 25
Hand in due: Thursday, June 8, 18:00
Direct inquiries via the ILIAS forum or to your respective tutor at:
caroline.jachmann@uni-tuebingen.de
meret.haeusler@student.uni-tuebingen.de
simon.hackl@uni-tuebingen.de
carolin.schwitalla@qbic.uni-tuebingen.de

Theoretical Assignments

1. **Proof of Cluster Distance Computation** (4P)

   To efficiently compute distances between clusters during the UPGMA algorithm, the update formula (see equation 6.4. on p. 88) was introduced in the lecture. In short: the distances between two clusters C_i and C_j are defined by the following equation

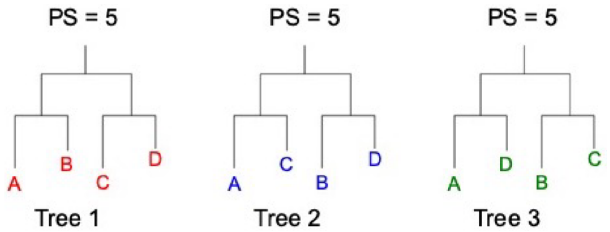
   $$d(i,j) = \frac{1}{|C_i||C_j|} \sum_{x \in C_i, y \in C_j} d(x,y)$$

   Let $C_k = C_i \bigcup C_j$, then prove that $d(k,l)$ for any l is given by

   $$d(k,l) = \frac{d(i,l)|C_i| + d(j,l)|C_j|}{|C_i| + |C_j|}$$

2. **Parsimony Score and MSA** (4P)

   For the following three phylogenetic trees on 4 taxa A, B, C, D with the respective parsimony scores, set up a minimal nucleotide multiple sequence alignment (with respect to the number of columns) $\mathbf{A}^*$ for each tree that result in the respective parsimony score $PS(T, \mathbf{A}^*)$ as shown below:

   With your solution of the MSA also prove that your MSA achieves the given parsimony score (using Fitch's algorithm).

Practical Assignments

3. Phylogenetic Analysis of SQR Sequences (6P)

For this exercise we ask you to use the phylogenetic software package MEGA11 `http://www.megasoftware.net/` (note that this software package needs a registration, but it is free for academic users).

Sulfide quinone reductase (SQR) is an enzyme critical for the growth of photo- and chemolitoautotrophic bacteria and archaea that use sulfide as an electron donor. It is part of the electron transport chain and catalyzes the first step of sulfide oxidation. In this process, electrons are transferred from SH2 to FAD and in the next step to quinone and finally used to reduce NAD+ to NADH. The SQR is a membrane-bound protein. In most organisms studied, it is integrated into the membrane; Only in *Rhodobacter capsulatus* it is known to be relatively easily separable from the membrane and is probably only superficially bound to the membrane on the extracellular side. Hydrogen sulfide enters the cells from the outside and thus directly into the SQR. The resulting elemental sulfur is deposited outside the cells. SQR also occurs in eukaryotes. However, the origin of eukaryotic SQRs is not clear, so in this task we ask you to examine SQR sequences with phylogenetic analysis.

(a) We provide you with a set of sequences at /resources/sequences.fasta. As a first step, we ask you to conduct a multiple sequence alignment of the provided sequences using ClustalW as provided by MEGA.

(b) Next, choose a Neighbor-Joining and Maximum-Parsimony based method to reconstruct two phylogenetic trees of your generated MSA.

(c) Summarize and discuss your results: Report your results and provide figures of the reconstructed trees with an informative caption as well as the name of the methods and the precise parameters used for their reconstruction. In your text, make sure that you correctly refer to the included figures.

Discuss whether the different methods agree on the tree or maybe only in parts. Is there any indication of a common ancestor of eukaryotic SQR (i.e., by clusters in the respective trees)? If yes, can you think of any explanation for this evolutionary event? In total, write maximally 400 words.

4. Cophenetic Correlation Coefficient (6P)

After a phylogenetic tree has been computed for a given distance matrix using a (distance) method of choice, one often would like to compute how well the reconstructed tree reflects the input data. One possibility is the so-called cophenetic correlation coefficient (CCC). The CCC takes two distance matrices as input, one is the original distance matrix and one is the derived distance matrix from a computed tree (called patristic or cophenetic distances, see p. 85 of the lecture notes). The CCC is then computed as

$$c(D,T) = \frac{\sum_i \sum_j (D_{ij} - \overline{D})(T_{ij} - \overline{T})}{\sqrt{\sum_i \sum_j (D_{ij} - \overline{D})^2 \sum_i \sum_j (T_{ij} - \overline{T})^2}}$$

where

- D_{ij} are the input distances between objects i, j in D.
- T_{ij} are the cophenetic (patristic) distances between leaves i, j in T.
- $\overline{D}$ and $\overline{T}$ are the average distances of D and T, respectively.

A CCC = 1 states that the computed tree perfectly reflects the input distances. A value close to 1 is a very good solution, while a value close to 0 reflects a random solution.

(a) Implement your own method that computes the CCC of two distance matrices. You are not allowed to used any library that provides a direct computation of the CCC. You can reuse your implementation of the last assignment for parsing scoring matrices.

(b) Read the original matrix distances_original.dist, as well as the two derived distance matrices distances_tree_1.dist and distances_tree_2.dist. Apply your method to evaluate the computation of the derived distance matrices distances_tree_1.dist and distances_tree_2.dist with respect to the original matrix distances_original.dist. Print the CCC value for each comparison to console with the name of the matrices.

Ensure that your program is submitted and executable as a JAR file. Indicate in your pdf report how to execute your program. The distance matrix files for this task are provided in the /resources directory of the assignment material.

Please read the questions carefully. If there are any questions, you may ask them during the tutorial session or in the forum of ILIAS. You will usually get an answer in time, but late e-mails (e.g. the evening of the hand-in) might not be answered in time. Please upload all your solutions to ILIAS. Don't forget to put your names on every sheet **and** in your source code files. Please pack both your source code as well as the theoretical part into one single archive file and give it a name using this scheme: `<name1>_<name2>_<Assignment>_<#>.zip`. The program should run without any modification needed.