Exercises – Phylogenetics

Universität Bielefeld, WS 2015/2016,
Dipl.-Inform. Damianos Melidis · B. Sc. Kevin Lamkiewicz
http://wiki.techfak.uni-bielefeld.de/gi/Teaching/2015winter/Phylogenetik

Exercise List 13 — 02.02.2016

Due to: 09.02.2016

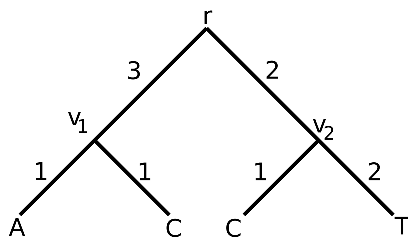
Exercise 1 Computing the Likelihood of a given tree. (4 Points)

Consider the given transition probability matrix $P(t)$ and the given tree T. Compute the *likelihood* of T. Considering your results, how are the nodes v_1 and v_2 of T labeled?

$$P(t) = \begin{pmatrix} 1-3a_t & a_t & a_t & a_t \\ a_t & 1-3a_t & a_t & a_t \\ a_t & a_t & 1-3a_t & a_t \\ a_t & a_t & a_t & 1-3a_t \end{pmatrix}$$

where

$$a_t = \frac{1 - exp(-4t/30)}{4}$$



Exercise 2 Pulley Principle. (4 Points)

The *Pulley Principle* from Joseph Felsenstein says, that the likelihood for a phylogenetic tree is independent of the location of the root node.
Consider the alphabet $\{A, G\}$ and a tree that contains two leaves, that are connected by the root node. Let leave M be labeled with A and leave N be labeled with G. The edge from M to the root has length t_M and the other edge length t_N.
Show that the likelihood for that tree is independent of the exact location of the root node (therefore it is only dependent of the sum of the lengths: $t_M + t_N$)

(a) Write down the likelihood in dependence of t_M and t_N. (Example: lecture notes page 80 upper figure.)

(b) Use the assumption that the process of evolution is *reversible* to write the likelihood in a way such that it only contains one π_i (e.g. π_G).

(c) Since P is a stochastic matrix we can use the *Chapman-Kolmogorov Equation*: $P(t_M + t_N) = P(t_M)P(t_N)$. You can simplify the likelihood with one of the four entries of $P(t_M + t_N)$ such that it is only dependent of the sum $t_M + t_N$.

Exercise 3 Jackknifing (3 Points)

Write the proceeding of the *delete-half jackknifing* as pseudo code (5–10 lines).
Let M be the method that reconstructs the tree T for a given multiple alignment A with length N: $T \leftarrow M(A)$. You can use the terms *edge* and *split* synonymously.

- Input: Method M, Alignment A, Tree T, number of replicates R.
- Output: Bootstrap-Support of every edge u in tree T.