

# Exercises – Phylogenetics

Universität Bielefeld, SS 2018

Dr. Roland Wittler

<https://gi.cebitec.uni-bielefeld.de/Teaching/2018summer/Phyloenetik>

## Exercise Sheet 8 — 14.06.2018

Due: 21.06.2018

### Task 1 Neighbor Joining.

(4 points)

Reconstruct a *neighbor joining* tree from the following matrix. Write down all steps.

	A	B	C	D	E
A:	0	8	5	8	2
B:		0	8	6	6
C:			0	8	4
D:				0	6
E:					0

### Task 2 Programming exercise: Splittree.

(4 points)

Implement a function that calculates the *isolation index* for the *split decomposition*: For a given distance matrix  $d$  on an object set  $O$ , calculate  $\alpha_{J,K}$  for a split  $J,K$  of  $O$ , i.e.,  $J \cap K = \{\}$  and  $J \cup K = O$ :

$$\alpha_{J,K} = \frac{1}{2} \min_{\substack{i,j \in J \\ k,l \in K}} (\max\{d_{ij} + d_{kl}, d_{ik} + d_{jl}, d_{il} + d_{jk}\} - d_{ij} - d_{kl})$$

For 4 bonus points implement also

- (a) a function that calculates the *split metric*: (1\* point)

$$\delta_{J,K}(i,j) = \begin{cases} 0 & \text{if } i \text{ and } j \text{ both are in } J \text{ or both are in } K \\ 1 & \text{otherwise} \end{cases}$$

- (b) a function that calculates the matrix  $d^1$ :  $d_{ij}^1 = \sum_{\text{splits } J,K} \alpha_{J,K} \delta_{J,K}(i,j)$ . (2\* points)

You do not need to calculate which splits are really valid ( $\alpha_{J,K} > 0$ ). Just sum over all possible splits. Even though this procedure is not efficient, it leads to the correct solution since  $\alpha_{J,K} = 0$  for all splits that are invalid.

Hint: For the representation/enumeration of subsets, numbers in binary format can be used. For example the set  $\{B, D, E\}$  can be represented by 01011.

- (c) a function that calculates the *splittable percentage*  $\rho$ : (1\* point)

$$\rho := \left( \frac{\sum_{\text{taxa } i,j} d_{ij}^1}{\sum_{\text{taxa } i,j} d_{ij}} \right) \cdot 100\%$$

Send your program by email and describe how to use the program. Make it as easy as possible to calculate all the  $\alpha$  and  $\rho$  in Task 3. A further matrix and some solutions are provided on the back of this sheet to test your implementation.

Please turn over! Bitte wenden!

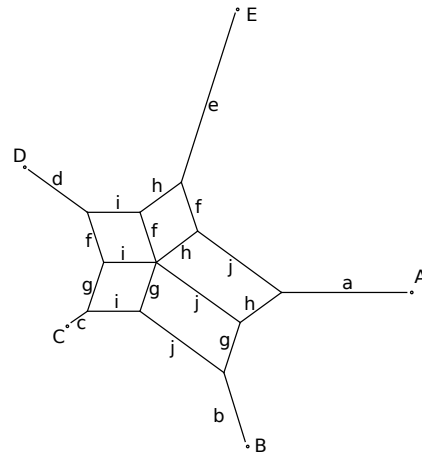
**Task 3 Splitstree.**

**(4 points)**

Solve this exercise with your implementation. (You can also use the software *Splitstree* ([www.splitstree.org](http://www.splitstree.org)) — describe the format of your input and your proceeding.)

Consider the following distance matrix and the corresponding splitstree:

	A	B	C	D	E
A:	0	6	8	9	9
B:		0	5	8	10
C:			0	4	8
D:				0	7
E:					0



- (a) Calculate the length of the edges *a* to *j* of the given network.

(Hint:  $a = \alpha_{\{A\},\{B,C,D,E\}}$ ,  $g = \alpha_{\{B,C\},\{A,D,E\}}$ , etc.)

- (b) The *Neighbor Joining* algorithm on the same distance matrix gives the tree

$$\left( \left( \left( C : \frac{7}{4}, D : \frac{9}{4} \right) : \frac{3}{4}, E : \frac{19}{4} \right) : \frac{7}{4}, \left( A : \frac{14}{4}, B : \frac{10}{4} \right) \right);$$

Compare this tree with the result of Task (a). Hint: Even if you have not solved Task (a), you can nevertheless compare both trees since the edge lengths in the image of the splitstree are scaled to the real edge lengths.

Matrix and some solutions to test your implementation for Task 2 (optional!):

	A	B	C	D	E
A:	0	6	8	5	10
B:		0	5	8	10
C:			0	4	8
D:				0	7
E:					0

J	K	$\alpha_{J,K}$
{A, B, C, E}	{D}	0.5
{A, B, D, E}	{C}	0.5
{A, B, E}	{C, D}	0.5
{A, C, D, E}	{B}	1.5
{A, C, E}	{B, D}	0.0
{A, D, E}	{B, C}	1.5
{A, E}	{B, C, D}	0.0
{B, C, D, E}	{A}	1.5
{B, C, E}	{A, D}	1.0
{B, D, E}	{A, C}	0.0
{B, E}	{A, C, D}	0.0
{C, D, E}	{A, B}	2.0
{C, E}	{A, B, D}	0.0
{D, E}	{A, B, C}	0.0
{E}	{A, B, C, D}	4.5

$$\sum_{\text{taxa } i,j} d_{ij}^1 = 64 \text{ (or 128 if you sum over both } (i,j) \text{ and } (j,i)).$$

$$\sum_{\text{taxa } i,j} d_{ij} = 71 \text{ (or 142, respectively.)}$$

$$\rho \approx 90,141\%$$