PHYLOGENY – TME3

2025-2026

Vinh-Son PHO
vinh-son.pho@sorbonne-universite.fr

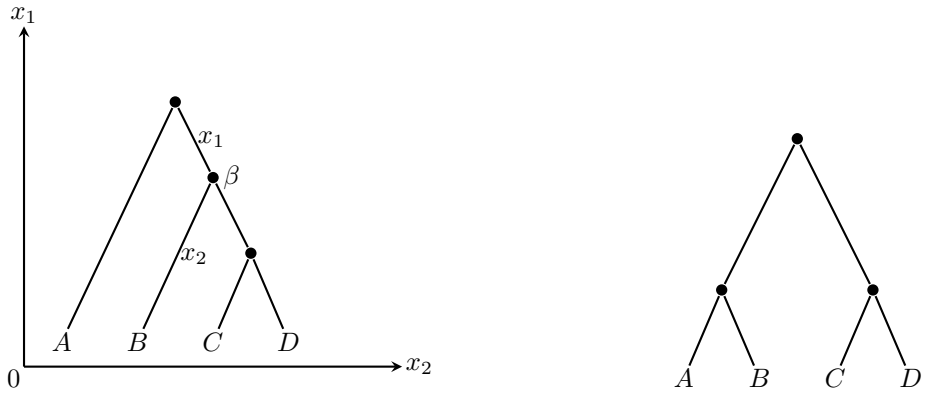
20 November 2025

General rules

- Reports must be sent by e-mail, using the subject "`[PHYG] TME3`", including in the body the names of the persons who worked on it (maximum two students per group). The deadline is 27th of November.
- Multiple files should be grouped in a compressed archive (`.tar.gz` or `.zip`).
- Your report *must be* in PDF format and named `student1_student2_TME4.pdf`. It should be simple, clear and well organized. Answers should be given in an exhaustive manner. Consider adding at the beginning a summary indicating the page of each answer.
- Source code must be well explained, commented and, most importantly, it should work without errors. Provide all needed information (*e.g.,* compiler/interpreter version) in a `README` file.
- All required materials can be found in the repository `https://github.com/20sxn/PHYG2025`.

Exercise 1

1. Given the tree topology T_1 depicted in Figure 1a, let x_1 be the distance between the root and the inner node β, and x_2 the distance between, β and B, we note $(T_1, (x_1, x_2))$ the set of trees that we can obtain by variying (x_1, x_2), assuming all other distances are 1. Draw the trees obtained when $x_1 = 0$, $x_2 = 0$ and $x_1 = x_2 = 0$.
2. Given the tree topology T_2 in Figure 1b, place x_3 and x_4 on T_2 so that $|(T_1, (x_1, x_2)) \cap (T_1, (x_3, x_4))| = 1$. Describe the intersection.



(a) (b)

Figure 1

3. Given the tree topology T_2 in Figure 1b, place x_3 and x_4 on T_2 so that $|(T_1, (x_1, x_2)) \cap (T_1, (x_3, x_4))| > 1$. Describe the intersection.

Note : the following exercises use the LG model, but they can also be done using the WAG model (or other substitution models, see `https://iqtree.github.io/doc/Substitution-Models`) with some modifications.

Exercise 2 : Maximum-Likelihood Distance estimation

The LG model assumes the independence of evolution between sites. Thus, the probability that sequence S evolves to sequence S' equals the product, over all sites i, of the probability that S_i evolves to S'_i. Hence, we will only focus on the evolution of a single site to lighten the notations for now. The second assumption is about the Markovian nature of site evolution : LG assumes that evolution has no memory, is time-continuous and time-homogeneous. The LG model is a continuous-time Markov chain in which the set of state corresponds to the characters in the sequences (i.e. amino acids). Let $P(t) = P_{x,y}(t)$ the matrix of substitution probabilities, where $P_{x,y}(t)$ is the probability of observing a substitution from x in one sequence to y in the other, with an elapsed time t between the 2 sequences. In the LG model a transition rate matrix Q is used to determine $P(t)$, with the equation :

$$P(t) = e^{Qt}$$

with $e^{\cdot}$ noting the matrix exponentiation which can be computed in python using `scipy.linalg.expm`.

1. Download the LG model `http://www.atgc-montpellier.fr/download/datasets/models/lg_LG.PAML.txt` in PAML format. The triangular matrix is noted R and the bottom row is noted π.

Amino acid are in this order : "ARNDCQEGHILKMFPSTWYV".
Derive Q using the following equations:

$$Q'_{x,y} = \pi_y R_{x \longleftrightarrow y} \text{ for } x \neq y$$
$$Q'_{x,y} = -\sum_{x \neq y} Q'_{x,y}$$
$$\mu = -\sum_{x} \pi_x Q'_{x,x}$$
$$Q = \frac{1}{\mu} Q' \text{ That way, the unit of time } t \text{ is substitution per site.}$$

You can check that the rows of e^{10000Q} are equal/close to the π vector.

2. Define a function to compute the log-likelihood of 2 aligned sequences S and S' with N ungapped positions and given a distance t (ignore positions that contain gaps) :

$$p_{a,b}(t) = \sum_{x} \pi_x P_{x,a}(\frac{t}{2}) P_{x,b}(\frac{t}{2}) \text{ where } x \text{ represent a hypothetical common ancestor}$$
$$\log L(t) = \sum_{i}^{N} \log p_{S_i, S'_i}(t)$$

3. Why do we need to use the log-likelihood instead of the likelihood ?

4. For the first 2 sequences in `danio_rerio_opsins.aln-fasta`, plot the Log-likelihood for $t \in [0, 2]$. Comment the curve ?

5. Compute the optimal distance between all pairs of sequences in `danio_rerio_opsins.aln-fasta` using Brent's method implemented in `scipy.optimize.minimize_scalar`. Note : this function returns a local minimum, but we want to maximize the log-likelihood.

6. Explain the principle behind Minimum Evolution.

7. Describe briefly how you could use NJ and NNI to build a Minimum Evolution tree starting from the previously computed distances.

Exercise 3 : Felsenstein's pruning algorithm

Algorithm 1 site_log_likelihood_per_state

Input: Node n (root of current subtree),
Site index s,
Multiple sequence alignment MSA,
Substitution rate matrix Q,
Equilibrium frequencies π,
Output: Vector ℓ_n of log-likelihoods for all possible amino acids at node n

if *n is a leaf* **then**
  Let $a \leftarrow \text{MSA}[n][s]$
  Initialize $\ell_n[i] \leftarrow -\infty$ for all states i
  $\ell_n[a] \leftarrow 0$; `// log(1)`
  return ℓ_n
end
foreach *child c of n* **do**
  Compute transition matrix $P_c \leftarrow \exp(Q \cdot t_c)$; `// $t_c$: branch length`
  $\log P_c \leftarrow \log(P_c + \varepsilon)$; `// $\varepsilon$ to avoid NaN`
  $\ell_c \leftarrow \text{site\_log\_likelihood\_per\_state}(c, s, \text{MSA}, Q, \pi)$
  foreach *state i at node n* **do**
    $m_c[i] \leftarrow \log \sum_j \exp(\ell_c[j] + \log P_c[j, i])$; `// use scipy.special.logsumexp`
  end
  Store m_c for this child
end
foreach *state i at node n* **do**
  $\ell_n[i] \leftarrow \sum_{\text{children } c} m_c[i]$
end
return ℓ_n

Algorithm 2 site_log_likelihood

Input: Node n (root of the tree),
Site index s,
Multiple sequence alignment MSA,
Substitution rate matrix Q,
Equilibrium frequencies π,
Output: Log-likelihoods of the tree n at a given site s

Let $logL\_vec \leftarrow \text{site\_log\_likelihood\_per\_state}(n, s, \text{MSA}, Q, \pi)$
Let $logL \leftarrow \log \sum_i \pi_i \exp(logL\_vec[i])$; `// use scipy.special.logsumexp`
return $logL$

Algorithm 3 msa_log_likelihood

Input: Node n (root of current subtree),
Multiple sequence alignment MSA,
Substitution rate matrix Q,
Equilibrium frequencies π,
Output: Vector ℓ_n of log-likelihoods for all possible amino acids at node n

```
Let total_logL ← 0

foreach position s of MSA do
    if MSA[s] does not contain gaps then
        site_logL ← site_log_likelihood(n, s, MSA, Q, π)
        total_logL ← total_logL + site_logL
    end
end
return total_logL
```

1. Explain briefly why we need to use the LogSumExp function when working with Log-likelihood

2. Implement the Felsenstein's pruning algorithm (Algorithm 1), using the Log-likelihood. Newick trees can be parsed and manipulated using `ete3`.

3. Implement Algorithms 2 and 3. Compute the log-likelihood of the tree (`danio_rerio_opsins.aln-fasta.treefile`) given the LG model and the MSA (`danio_rerio_opsins.aln-fasta`).

4. Describe how to modify Algorithm 3 to predict the most likely ancestral sequence at the root, according to the LG model (ignoring gapped positions). Compute the most likely ancestral sequence.

5. (optional) Use IQ-tree or RaxML to build trees better with more complex models based on the alignment. What log-likelihood do these methods get ?

References

[1] Le, Si Quang, Gascuel, Olivier. "An Improved General Amino Acid Replacement Matrix." Molecular Biology and Evolution 25.7 (2008): 1307–1320.

[2] Felsenstein, Joseph. "Maximum Likelihood and Minimum-Steps Methods for Estimating Evolutionary Trees from Data on Discrete Characters" Systematic Biology 22.3 (1973): 1307–1320.

[3] Minh, Bui Quang, et al. "IQ-TREE 2: New Models and Efficient Methods for Phylogenetic Inference in the Genomic Era" Molecular Biology and Evolution 37.5 (2020): 1530–1534.

[4] Stamatakis, Alexandros. "RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies" Bioinformatics 30.9 (2014): 1312–1313.