

# Introduction aux Processus Stochastiques

## Projet Chaîne de Markov en temps discret

### Inférence d'arbres phylogénétiques

Profs.: Céline ESSER et Pierre GEURTS

Année académique 2024-2025

Ce travail est à réaliser par groupe de 2 étudiants. Le rapport et le code source sont à remettre via Gradescope pour le **vendredi 9 mai 2025 à 23h59** au plus tard.

## Contexte général et objectifs

L'objectif pédagogique de ce projet est de manipuler des chaînes de Markov en temps discret dans le cadre d'une application réelle en phylogénie. La phylogénie est une sous-discipline de la biologie qui étudie les liens de parenté entre êtres vivants (au sens large). Elle vise à reconstituer l'évolution des organismes vivants. L'outil de prédilection du domaine est l'arbre phylogénétique. On se focalisera ici sur le problème de l'inférence d'un arbre phylogénétique entre plusieurs êtres à partir de leur séquence d'ADN (ou un morceau de cette séquence).

Le projet est divisé en trois parties. Dans la première partie, on se focalise sur l'utilisation de chaînes de Markov pour la modélisation de l'évolution d'une séquence d'ADN. Dans la seconde partie, on traitera du problème de la représentation de l'évolution des séquences de plusieurs êtres vivants par un arbre s'appuyant sur ce type de modèle. Enfin, dans la troisième partie, on abordera le problème de l'inférence d'un arbre à partir d'un ensemble de séquences.

*Remarque : les outils qui sont traités dans ce projet sont des version simplifiées d'outils et d'algorithmes réellement utilisés dans le domaine de la phylogénie [1]. Pour plus de détails, vous pouvez consulter d'autres sources, notamment celles données dans la section Références.*

## Partie 1 : modèle d'évolution de l'ADN

Une séquence d'ADN<sup>1</sup> consiste en une succession de nucléotides pouvant chacun avoir l'une parmi quatre valeurs dans l'ensemble  $\{a, c, g, t\}$ <sup>2</sup> Dans la suite, on notera ces valeurs de 1 à 4 dans l'ordre alphabétique  $a, c, g, t$ , et on utilisera des caractères minuscules pour les représenter.

---

1. Acide désoxyribonucléique.

2. a pour adénine, c pour cytosine, g pour guanine, t pour thymine.

Au fil de l'évolution, une séquence d'ADN subit des modifications ponctuelles. Bien qu'il y ait d'autres mécanismes d'évolution de l'ADN, on se concentrera dans ce projet uniquement sur les mutations, c'est-à-dire une modification d'un nucléotide de la séquence en un autre. Les modèles courants de l'ADN font les hypothèses suivantes :

- l'apparition de mutations dans une séquence ne dépend que de la séquence elle-même et pas des mutations précédentes qu'elle a subies (le processus de mutation est sans mémoire).
- L'apparition d'une mutation en une position de l'ADN est indépendante des autres positions dans la séquence et toutes les positions évoluent selon le même processus.

Vu ces hypothèses, il est naturel de modéliser les mutations à une position de l'ADN par une chaîne de Markov  $CM(\mathbf{u}, \mathbf{P})$  à quatre états, un par nucléotide, où  $\mathbf{u}$  représente la probabilité initiale d'apparition d'un nucléotide donné dans la séquence et  $P_{ij}$  représente la probabilité de passer du nucléotide  $i$  au nucléotide  $j$  à chaque pas de temps, ce pas de temps représentant une unité de temps arbitraire d'évolution. On appelle un modèle de ce type, un modèle de substitution<sup>3</sup>. Etant donnée la première hypothèse ci-dessous, ce modèle définit à son tour une chaîne de Markov expliquant les mutations sur des séquences d'ADN. Si on note  $l$  la longueur des séquences, cette chaîne de Markov, notée  $CM(\mathbf{u}_S, \mathbf{P}_S)$ , possède  $4^l$  états et sa distribution initiale et sa matrice de transition peuvent être dérivées directement du modèle de substitution  $CM(\mathbf{u}, \mathbf{P})$ . Dans la suite, on notera  $(S_n)_{n \in \mathbb{N}} \sim CM(\mathbf{u}_S, \mathbf{P}_S)$  le processus stochastique à temps discret modélisant l'évolution des séquences.

La plupart des modèles de substitution utilisés sont des chaînes régulières et telles que  $\mathbf{u}$  soit l'unique distribution stationnaire de  $P$ . Une hypothèse supplémentaire généralement appliquée à la chaîne est que celle-ci soit *réversible*. Une chaîne  $CM(\mathbf{u}, \mathbf{P})$  est réversible si pour tout  $i, j$  :

$$u_i p_{ij} = u_j p_{ji}. \quad (1)$$

Un modèle classique que nous utiliserons dans la suite est le modèle de Kimura à deux paramètres ( $\alpha > 0$  et  $\beta > 0$ ) :

$$\mathbf{P}_{kim} = \begin{pmatrix} 1 - \alpha - 2\beta & \beta & \alpha & \beta \\ \beta & 1 - \alpha - 2\beta & \beta & \alpha \\ \alpha & \beta & 1 - \alpha - 2\beta & \beta \\ \beta & \alpha & \beta & 1 - \alpha - 2\beta \end{pmatrix}, \quad (2)$$

$$\mathbf{u}_{kim} = (0.25 \quad 0.25 \quad 0.25 \quad 0.25). \quad (3)$$

La deux paramètres  $\alpha$  et  $\beta$  permettent de moduler le taux de substitution en fonction des deux groupes  $\{a, g\}$  et  $\{c, t\}$  de nucléotides, appelées respectivement les purines et les pyrimidines. La probabilité  $\alpha$  représente la probabilité d'une substitution au sein du même groupe, purines ou pyrimidines, et  $\beta$  la probabilité de substitution entre groupes. Le premier type de substitution s'appelle une mutation de transition, alors que le second type s'appelle une mutation de transversion. Il y a généralement plus de transitions que de transversions (de l'ordre de deux fois plus dans le génome humain). La chaîne ainsi définie est trivialement réversible puisque la matrice de transition est symétrique et la distribution initiale est uniforme mais tout modèle de substitution réversible ne correspond pas nécessairement à une matrice de transition symétrique. Pour la question 3, ci-dessous, on vous fournit sur Ecampus une chaîne réversible, correspondant alors à une distribution initiale non uniforme sur les nucléotides.

3. [https://en.wikipedia.org/wiki/Substitution\\_model](https://en.wikipedia.org/wiki/Substitution_model)

**Questions :** Dans votre rapport, répondez aux questions suivantes :

1. Soient deux séquences d'ADN,  $s$  et  $s'$ , de longueur  $l$ . Expliquez comment calculer à partir du modèle de substitution  $CM(\mathbf{u}, \mathbf{P})$  sur les nucléotides et sans calculer explicitement la matrice de transition des séquences  $\mathbf{P}_S$ , la probabilité  $\mathbb{P}(S_t = s' | S_0 = s)$  de passer de  $s$  à  $s'$  en  $t$  pas de temps, ainsi que la probabilité conjointe  $\mathbb{P}(S_0 = s, S_t = s')$ .
2. Prouvez que si le modèle de substitution est une chaîne de Markov réversible alors,  $\mathbb{P}(S_0 = s, S_t = s') = \mathbb{P}(S_0 = s', S_t = s)$ .
3. Vérifiez ce résultat pour les séquences *ccat* et *ccgt* pour  $t = 5$  et la chaîne réversible de matrice de transition non symétrique fournie sur Ecampus.
4. Les biologistes sont intéressés par la distance génétique entre deux séquences d'ADN définie par le nombre de mutations subies par la première pour se transformer en la deuxième. Cette distance peut être approximée par le nombre de nucléotides qui diffèrent entre les deux séquences. Cette approximation sous-estime cependant la distance génétique réelle parce qu'elle ne tient pas compte du fait que plusieurs mutations successives d'un même nucléotide peuvent le ramener à sa valeur initiale. Pour vérifier l'ampleur de cette sous-estimation, générez une séquence de longueur  $l = 1000$  et faites évoluer cette séquence sur 1500 pas de temps selon le modèle de Kimura avec  $\alpha = 2\beta = 0.0008$  et tracez sur une courbe l'évolution du nombre de différences entre la séquence à chaque pas de temps et la séquence initiale en fonction du nombre de mutations observées réellement sur ce nombre de pas de temps. Moyennez cette courbe sur 10 expériences. Expliquez vers quelle valeur cette courbe converge.
5. Une autre mesure de distance entre séquences, qui est celle que nous utiliserons pas la suite, est le nombre de pas de temps d'évolution qui les séparent. Une manière d'estimer ce nombre de pas de temps est d'utiliser la valeur  $\hat{t}_{ML}$  qui maximise la probabilité jointe des deux séquences prenant en compte leur espacement :

$$\hat{t}_{ML} = \arg \max_t \mathbb{P}(S_0 = s, S_t = s'). \quad (4)$$

En statistique, on appelle cet estimateur un estimateur au maximum de vraisemblance,  $\mathbb{P}(S_0 = s, S_t = s')$  étant une mesure de la vraisemblance de la valeur de  $t$ . Générez une séquence  $s$  de longueur  $l$  et faites la évoluer selon un modèle de Kimura avec  $\alpha = 2\beta = 0.01$  sur un nombre de pas de temps fixé  $t^*$  (10 par exemple) pour obtenir  $s'$ . Faites varier  $l$  (par exemple dans l'ensemble  $\{10, 50, 100, 500, 1000\}$ ) et vérifiez si la maximisation de (4) permet de retrouver  $t^*$ . Répétez si nécessaire plusieurs fois les expériences. Commentez vos résultats.<sup>4</sup>

## Partie 2 : arbre phylogénétiques

Un arbre phylogénétique est un modèle qui permet de représenter l'historique d'évolution de plusieurs séquences. On notera ces séquences, en nombre  $N$ ,  $\mathcal{S} = \{s_1, \dots, s_N\}$  et on notera  $l$  leurs longueurs. Un arbre phylogénétique est un arbre binaire<sup>5</sup> entier (chaque nœud interne

4. Quand les séquences sont longues, le calcul direct de  $\mathbb{P}(S_0 = s, S_t = s')$  va mener à des valeurs numériques très petites, ce qui risque de poser des problèmes de précision. Il est préférable en général de calculer plutôt  $\log \mathbb{P}(S_0 = s, S_t = s')$ , qu'on appelle parfois la log-vraisemblance, en remplaçant le produit des probabilités sur les  $l$  nucléotides par une somme des logarithmes de ces probabilités.

5. Il est possible de permettre à des nœuds d'avoir plus de deux successeurs mais on se limitera ici, sans perte de généralité, à des arbres binaires.

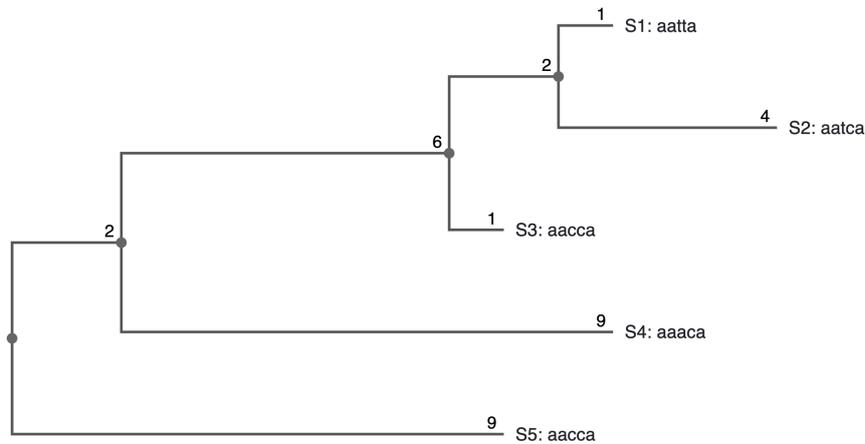


FIGURE 1 – Un exemple d’arbre phylogénétique défini sur 5 séquences. Dans cette représentation classique, la racine se trouve à gauche et le temps s’écoule vers la droite. Chaque nœud, sauf la racine, est étiqueté par la longueur de la branche qui le connecte à son parent.



FIGURE 2 – Deux arbres simples définis sur les mêmes séquences.

possède soit zéro, soit deux fils). Chaque feuille de l’arbre est étiquetée par une des  $N$  séquences et chaque nœud interne (il y en a  $N - 1$ ) correspond à une séquence non observée qui est supposée être l’ancêtre de ses descendants dans l’arbre. Chaque branche est étiquetée par une longueur qui représente le nombre de pas de temps d’évolution qui séparent un parent du fils correspondant à la branche. La figure 1 représente un exemple d’arbre phylogénétique défini sur 5 séquences<sup>6</sup>. Etant donné un modèle de substitution  $CM(\mathbf{u}, \mathbf{P})$ , l’arbre est supposé avoir généré l’ensemble des séquences aux feuilles en partant d’une séquence aléatoire à la racine (dont les nucléotides sont tirés indépendamment les uns des autres selon  $\mathbf{u}$ ) et en utilisant le modèle de substitution à chaque branche pour générer, de la racine vers les feuilles, les séquences des fils gauche et droit à partir de la séquence de leur parent en prenant en compte la longueur de la branche.

Une question centrale de la phylogénie est de déterminer un arbre plausible à partir d’un ensemble de séquences, permettant ainsi de retracer l’histoire évolutive sous-jacente à ces séquences. Avant de passer à ce problème dans la partie suivante, les questions ci-dessous visent à vous faire comprendre les propriétés de cet arbre et à vous faire implémenter certaines fonctions utiles pour la suite.

### Questions :

1. Etant donné un arbre et un modèle de substitution, un problème intéressant est de déterminer la probabilité que les séquences aux feuilles aient été générées par l’arbre,

6. Cette représentation a été générée par l’outil en ligne <https://beta.phylo.io/>.

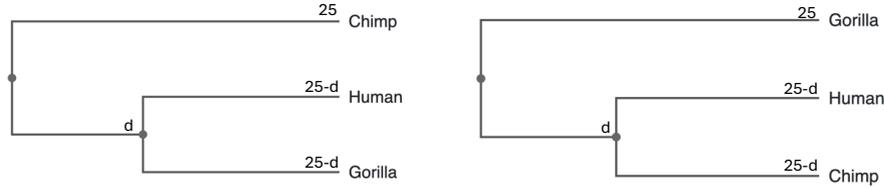


FIGURE 3 – Deux arbres potentiels à évaluer pour trois séquences de grands singes.

notée  $\mathbb{P}(\mathcal{S}|\mathcal{T})$ . On appelle cette probabilité la vraisemblance de l'arbre  $\mathcal{T}$  au vu des séquences. Soient les deux arbres de la figure 2 définis sur les mêmes séquences. En supposant un modèle de Kiruma avec  $\alpha = 2\beta = 0.01$ , déterminez pour les deux arbres, en détaillant votre calcul, la probabilité qu'ils aient généré les deux séquences aux feuilles. Comparez ces deux probabilités à la probabilité  $\mathbb{P}(S_0 = s_1, S_5 = s_2)$  calculée comme à la sous-question 3 de la partie 1. Expliquez ces résultats (qui illustre ce qu'on appelle l'effet poulie [2]).

2. Ecrivez un code générique pour calculer *le plus efficacement possible* la probabilité  $\mathbb{P}(\mathcal{S}|\mathcal{T})$  pour n'importe quel arbre et n'importe quel ensemble de séquences. Vous pouvez pour cela suivre la méthode décrite dans l'article [2], fourni sur Ecampus. Expliquez brièvement le principe de votre implémentation et donnez dans le rapport la valeur trouvée pour l'arbre et les séquences de la figure 1.
3. Soient les séquences (fictives) d'ADN de trois grands singes (fournies sur Ecampus) : chimpanzé, gorille et humain, de longueur 500. On aimerait déterminer si l'humain est plus proche du chimpanzé ou du gorille. Pour cela, on voudrait déterminer lequel des deux arbres à la figure 3 explique le mieux les trois séquences. On supposera un modèle de Kimura avec  $\alpha = 2\beta = 0.01$  et que les trois séquences sont séparées de leur ancêtre commun du même nombre de pas de temps fixé à 25. Déterminez pour chacun des modèles la valeur de  $d$  sur le graphe qui maximise la vraisemblance et ensuite l'arbre le plus vraisemblable. Sur base du résultat, répondez à la question initiale.

### Partie 3 : inférence d'un arbre par MCMC

Inférer un arbre plausible à partir d'un ensemble de séquences est un problème non trivial et de nombreux algorithmes ont été proposés dans la littérature pour le résoudre. L'approche qu'on cherchera à implémenter ici consistera à essayer de trouver l'arbre qui maximise la vraisemblance  $\mathbb{P}(\mathcal{S}|\mathcal{T})$ . Ce problème ne peut pas être résolu efficacement de manière exacte et une recherche exhaustive est extrêmement coûteuse : le nombre d'arbres différents, définis chacun par une structure et une combinaison des longueurs de branches, est en effet extrêmement important même pour un petit nombre  $N$  de séquences.

On se propose ici d'utiliser l'algorithme de Metropolis-Hastings (MH) pour résoudre le problème (voir [3, 4] pour des utilisations similaires). L'idée est de faire l'hypothèse que la vraisemblance définit une distribution de probabilité sur les arbres : distribution est en

$$P_{\mathcal{S}}(\mathcal{T}) = \frac{\mathbb{P}(\mathcal{S}|\mathcal{T})}{\sum_{\mathcal{T}'} \mathbb{P}(\mathcal{S}|\mathcal{T}')} \quad (5)$$

et d'ensuite utiliser l'algorithme MH pour construire une chaîne de Markov ayant pour distribution stationnaire cette distribution. Si on génère ensuite une réalisation suffisamment longue de cette chaîne, elle devrait générer des arbres tirés de cette distribution et donc, ce faisant, nous permettre de trouver des arbres de vraisemblances élevées.

L'algorithme aura le schéma général suivant :

1. On démarre d'un arbre  $\mathcal{T}_0$  arbitraire.
2. Pour  $t$  allant de 1 à  $T$  :
  - (a) On choisit aléatoirement un arbre  $\mathcal{T}'$  dans le voisinage de  $\mathcal{T}_{t-1}$ .
  - (b) On calcule :

$$\alpha \leftarrow \min \left( 1, \frac{P_{\mathcal{S}}(\mathcal{T}')}{P_{\mathcal{S}}(\mathcal{T}_{t-1})} \right) \quad (6)$$

- (c) On génère un nombre aléatoire  $u$  selon un loi uniforme sur  $[0, 1]$
- (d) Si  $u \leq \alpha$ ,  $\mathcal{T}_t \leftarrow \mathcal{T}'$ . Sinon,  $\mathcal{T}_t \leftarrow \mathcal{T}_{t-1}$ .

3. On renvoie l'arbre parmi  $\{\mathcal{T}_0, \mathcal{T}_1, \dots, \mathcal{T}_T\}$  qui maximise  $\mathbb{P}(\mathcal{S}|\mathcal{T})$ .

Pour l'étape 2.(a), on choisira un voisin de l'arbre courant aléatoirement par le biais de deux opérateurs modifiant respectivement la longueur des branches et la structure de l'arbre :

- Le premier opérateur choisit une branche au hasard et modifie son poids en lui ajoutant une valeur aléatoire choisie uniformément entre  $-k$  et  $+k$ .
- Le second opérateur choisit au hasard un nœud interne de l'arbre, excepté la racine, et échange un de ses deux fils, choisi au hasard également, avec son nœud frère.

Le deuxième opérateur est illustré à la figure 4. On choisira avec une probabilité  $p_{length}$  le premier opérateur et avec une probabilité  $1 - p_{length}$  le second, où  $p_{length}$  est un paramètre de l'algorithme à déterminer<sup>7</sup>.

*Remarque :* Etant donné l'effet poulie illustré dans la sous-question 1 de la partie 2, le nœud racine d'un arbre phylogénétique peut être replacé sur n'importe quelle branche de l'arbre sans changer sa vraisemblance par rapport aux séquences. Pour pouvoir associer un ordre chronologique selon les branches, il est courant d'inclure dans l'ensemble des séquences une séquence particulière, appelée l'"outgroup", que l'on sait devoir être connectée directement à la racine de l'arbre et à aucune autre séquence. Une fois l'arbre appris, on peut donc le réorienter pour imposer que ce nœud soit l'un des fils directs de la racine. Les autres branches seront alors dirigées depuis la racine de manière chronologique. On supposera ici que cet outgroup est connu a priori. La structure d'arbre initiale devra être telle que l'outgroup sera imposé comme le fils gauche de la racine et vous devrez implémenter l'opérateur d'altération de la structure pour ne jamais modifier cette partie de l'arbre (il suffira d'exclure le fils droit de la racine des nœuds internes pouvant être sélectionnés par l'opérateur de modification de la structure).

---

7. Cette manière d'explorer le graphe ne respecte pas exactement la contrainte du choix uniforme d'un voisin utilisée dans le cours. On peut cependant montrer que les propriétés de l'algorithme MH restent vérifiées pour autant que la probabilité d'aller d'un nœud  $i$  à un nœud  $j$  soit identique à la probabilité d'aller du nœud  $j$  au nœud  $i$ , ce qui est le cas ici.

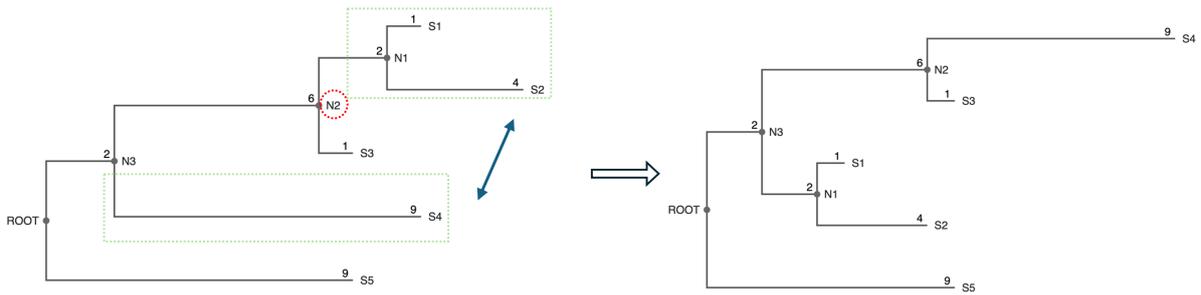


FIGURE 4 – Illustration de l’opérateur de modification de la structure. Le nœud interne choisi au hasard est le nœud  $N2$ . On a alors le choix entre échanger son sous-arbre de gauche (branche  $(N2, N1)$ ) et son sous-arbre de droite (branche  $(N2, S3)$ ) avec le sous-arbre de droite de  $N3$ . Dans l’exemple, c’est la première possibilité qui est choisie. Notez que les longueurs des branches sont inchangées suite à ces opérations (le nouvelle branche  $(N3, N1)$  garde la longueur de la branche  $(N2, N1)$ ).

**Questions** Pour cette partie du projet, on vous demande d’implémenter cet algorithme et de l’appliquer à un ensemble de 21 séquences fournies sur Ecampus. De manière fictive, on supposera que ces séquences correspondent à 21 variants d’un même virus récoltés à différents endroits dans le monde (indiqués dans le nom des virus). Même si les virus ne sont pas des êtres vivants, leur séquence d’ADN subit des mutations comme les cellules vivantes lors de leurs propagations et l’application de la phylogénie à ces séquences permet en principe de retracer la propagation du virus au cours du temps (ce principe a été utilisé par exemple pour étudier la propagation du COVID [5]). Pour ce problème, vous pourrez supposer que la longueur d’une branche de l’arbre ne peut pas excéder 10 et que le modèle de substitution est un modèle de Kimura avec les paramètres  $\alpha = 2\beta = 0.01$ . Vous pouvez utiliser la première séquence comme l’outgroup, qui correspond à la première souche du virus jamais découverte dont tous les autres variants devraient découler.

Dans votre rapport, on vous demande au minimum :

- de décrire brièvement votre implémentation de l’algorithme, en précisant en particulier comment vous avez déterminé la structure initiale de l’arbre,
- d’illustrer, notamment par des courbes d’évolution de la (log-)vraisemblance en fonction des itérations, le bon comportement de l’algorithme et en particulier de justifier vos choix pour les paramètres tels que la valeur de  $K$ , le nombre d’itérations  $T$ , la probabilité  $p_{length}$  régulant le choix de l’opérateur,
- de montrer dans le rapport le meilleur arbre finalement obtenu à partir des 21 séquences fournies et de le discuter. Quelles sont vos hypothèses quant à la manière dont le virus s’est propagé au cours du temps ?

Cette partie du projet est moins guidée. Pour le point 2, vous pouvez, en préambule à l’application sur les séquences fournies, tester les performances de l’algorithme sur des données générées par vous-mêmes à partir d’un arbre connu. Vous pouvez si vous le souhaitez également tester d’autres opérateurs de modification du graphe.

## Ressources

Toutes les ressources relatives au projet sont disponibles sur Ecampus. Pour vos expériences, nous vous encourageons à utiliser le Python mais vous pouvez utiliser un autre langage de programmation si vous le souhaitez (avec notre accord, si ce n'est pas C/C++, Java, Mathematica, ou R). Un code python de démarrage est fourni définissant une classe d'arbre basique et contenant toutes les séquences et les arbres présentés dans le rapport. Vous êtes libres de l'utiliser ou pas.

## Soumission

Vous devez nous fournir un rapport au format pdf contenant vos réponses, concises mais précises, aux questions posées ainsi que le code que vous avez utilisé pour y répondre. Le rapport et le code doivent être soumis de manière séparée sur Gradescope (le code cours sera fourni sur Ecampus en temps voulu). Les deux doivent être soumis pour le **vendredi 9 mai 2025 à 23h59** au plus tard.

**Plagiat et utilisation de l'IA.** Quel que soit le langage choisi, on vous demande d'implémenter les fonctions nécessaires pour répondre aux questions de l'énoncé par vous-mêmes. Vous ne pouvez pas utiliser une boîte à outils existante qui ferait partiellement ou totalement ce qui vous est demandé. De même, l'utilisation d'outils d'IA générative est interdite pour générer le code demandé ou pour générer directement les réponses aux questions posées dans l'énoncé. Vous pouvez cependant l'utiliser pour générer du code annexe non lié à la matière, par exemple pour générer des graphiques de résultats, ou pour des corrections grammaticales ou d'orthographe dans votre rapport, en respectant la charte d'utilisation des outils d'intelligence artificielle de la faculté<sup>8</sup>. En soumettant votre projet, vous confirmez avoir respecté ces règles. En cas de doute sur l'origine d'un code ou texte remis, vous serez convoqués pour obtenir des explications et le cas échéant, des sanctions seront prises.

Bon travail !

## Références

- [1] Denis Baurain. Notes du cours GENE0448 - méthodes de phylogénie, ULiège, March 2024.
- [2] Joseph Felsenstein. Evolutionary trees from dna sequences : a maximum likelihood approach. *Journal of molecular evolution*, 17 :368–376, 1981.
- [3] Bret Larget and Donald L Simon. Markov chain monte carlo algorithms for the bayesian analysis of phylogenetic trees. *Molecular biology and evolution*, 16(6) :750–759, 1999.
- [4] Shuying Li, Dennis K Pearl, and Hani Doss. Phylogenetic tree construction using markov chain monte carlo. *Journal of the American statistical Association*, 95(450) :493–508, 2000.
- [5] Erwan Sallard, José Halloy, Didier Casane, Jacques van Helden, and Étienne Decroly. Retrouver les origines du sars-cov-2 dans les phylogénies de coronavirus. *médecine/sciences*, 36(8-9) :783–796, 2020.

---

<sup>8</sup>. Disponible ici : [https://www.fsa.uliege.be/upload/docs/application/pdf/2023-11/chatgpt\\_student.pdf](https://www.fsa.uliege.be/upload/docs/application/pdf/2023-11/chatgpt_student.pdf).