L3 - 2016-2017

Introduction à la Bio-informatique

Projet Python

~ GSELL Louise ~

01-05-2017

Ce projet consiste en une étude automatisée de séquences protéiques et nucléiques. Dans le cadre de cette étude, il nous était demandé de nous intéresser plus particulièrement à la recherche d'îlots CpG dans les séquences nucléiques et à la recherche de régions hydrophobes dans les séquences protéiques.

Cette étude est permise par l'utilisation d'un script composé de six modules. Ce script permet à l'utilisateur de choisir le type de séquence qu'il souhaite étudier et d'étudier des fiches fasta en ligne aussi bien que des fichier fasta. Il lui permet également s'il le souhaite, de compléter ses résultats par des graphiques et d'approfondir son étude par une analyse similaire sur une ou plusieurs séquences aléatoires de même longueur et/ou de même composition. L'utilisateur à également la possibilité d'effectuer plusieurs analyses à la suite.

Je vous présenterai donc, dans un premier temps le script permettant d'effectuer ces analyses, son découpage en modules ainsi que les remarques et améliorations possibles le concernant. Puis dans un deuxième temps je commenterai l'intérêt biologique de ce projet ainsi qu'un ensemble d'exemples de résultats générés.

Explications et remarques sur le projet

- ### Choix de l'organisation des différents modules et présentations de leur contenu

La partie programme du projet est constituée d'un script contenant sept modules et d'un fichier texte, le « Read me ». Le script est composé d'un module (***analyse_sequence_fasta***) propre aux consignes de ce projet, qui appelle l'ensemble des cinq autres modules (directement ou indirectement). Ce module serait donc difficilement réutilisable dans un autre contexte. C'est donc sur ce critère que les trois seules procédures qu'il contient y sont regroupées. En effet, elles permettent d'obtenir un certain éventail de résultats qui est propre à ce projet. Les cinq autres principaux modules contiennent quant à eux des fonctions ou procédures qui pourraient être réutilisées dans d'autres projets visant à l'étude de séquences (ou de chaînes de caractères quelconques, pour certaines).

Cependant pour simplifier l'interface avec l'utilisateur, c'est avec un septième module ***module_de_lancement*** ayant pour seul rôle l'appel du programme, que le programme est lancé. Ainsi l'utilisateur n'est pas confronté à un ensemble de lignes de code incompréhensible pour lui lors du lancement du script.

1. Deux modules de récupération de séquence

Le module récupération de séquence : ***recuperation_sequence_fasta***, ne contient qu'une fonction (« *entree()* »). Il permet de récupérer une séquence au format fasta dans un fichier placé dans le répertoire courant, ou en ligne, si le poste de travail dispose d'une connexion internet. Pour ce faire, ce module fait appel au module ***lire_fasta*** qui contient une fonction permettant de lire les fiches en ligne et une autre permettant de lire les fichiers dans le répertoire courant. Le rôle de la fonction *entree()* est d'interagir avec l'utilisateur contrairement aux fonctions du module ***lire_fasta*** c'est pourquoi ces deux modules sont distincts. L'autre rôle majeur de cette fonction est de gérer les différentes erreurs pouvant survenir lors de l'interaction avec l'utilisateur, ou lors de l'appel des fonctions de ***lire_fasta***. De plus la fonction *entree()* est conçue pour ne s'interrompre que si l'utilisateur le souhaite ou que la séquence a été récupérée.

2. Un module d'analyse de séquence nucléique

Un autre module (***analyse_ADN***) regroupe l'ensemble des fonctions permettant d'étudier des séquences nucléiques. Entre autres, une fonction permettant de connaitre la composition de la séquence, une fonction permettant de connaitre le pourcentage de C+G et le nombre de CpG contenu dans la séquence entière ou dans chacune des fenêtres glissantes de taille donnée en entrée, et une fonction permettant de calculer ses deux valeurs plus le rapport CpG dans les cas cités précédemment. Ces deux fonctions paraissent un peu redondantes mais sont nécessaires à accélérer l'étude complète de la séquence en ne calculant que les résultats souhaiter dans chaque cas. En effet, elles sont bien plus rapides que si chaque valeur était calculée par des fonctions distinctes, puisque cela impliquerait de recalculer l'ensemble des fenêtres glissantes à chaque fois. Cependant les fonctions calculant uniquement une des valeurs précédemment citées sont également présentes dans le module pour permettre une utilisation indépendante à d'autres fins.

3. Un module d'analyse protéique

Un quatrième module (***analyse_proteine***) regroupe l'ensemble des fonctions permettant d'étudier des séquences protéiques. Notamment une fonction permettant de passer du code trois lettres au code une lettre (au cas où une séquence contenue dans un fichier soit codée en code trois lettres), une fonction permettant de calculer le nombre de résidus hydrophobes, le pourcentage de résidus chargés et la charge net de la séquence en ne la parcourant qu'une fois. Comme pour le module précédent les fonctions permettant de calculer ces valeurs séparément existent pour faciliter l'usage dans d'autre cas et ainsi éviter que le module ne soit spécifique au projet.

4. Un module de création de séquences aléatoires

Un cinquième module (***creation_seq_aleatoires***) permet la création de séquences aléatoires de même type et de même longueur et de séquence aléatoire de même composition que la séquence de référence prise en entrée. Ces deux fonctions sont donc utilisables aussi bien

pour des séquences protéiques que nucléiques. Si des « N » sont présents dans une séquence nucléotidique de référence ils seront remplacés dans la séquence aléatoire de même longueur par n'importe quel nucléotide.

5. Un module d'analyse séquence et de génération de dossiers de résultats

Le dernier module (***analyse_sequence_fasta***) qui fait appel à tous les autres est composé de trois procédures.

Une première procédure (« *resultat_ADN()* ») permet l'étude des séquences nucléiques et la génération de résultats. Elle crée des fichiers tabulés dans lesquels on trouve un premier tableau contenant les résultats de l'analyse de la séquence entière, et si l'analyse par fenêtre glissante a pu être effectuée, un deuxième tableau contenant les résultats de l'analyse par fenêtre. L'analyse de la séquence entière permet de récupérer le pourcentage de cytosine et de guanine de la séquence mais également le pourcentage de CpG (di nucléotides CG, cytosine suivit de guanine) et la composition de la séquence en chaque élément. Quant à l'analyse par fenêtre elle permet de calculer pour chaque fenêtre le pourcentage de cytosine et de guanine, le nombre de CpG, le rapport CpG (CpGnbre_observé / CpGnbre_attendu avec CpGnbre attendu = (Cnbre observé * Gnbre observé) / Taille de la fenêtre), ce qui permet de déduire la présence ou l'absence d'un îlot CpG. Elle permet également de créer des graphiques représentant les résultats de l'analyse par fenêtre. Cette procédure fait donc appel au module ***analyse_ADN*** et au module ***matplotlib*** si elle doit tracer des graphiques. La seconde procédure (« *resultat_prot()* ») fonctionne de façon très similaire mais fonctionne sur des séquences protéiques. Elle génère le même type de résultats, cependant l'analyse de la séquence entière permet de calculer le nombre d'acides aminés hydrophobes, le pourcentage d'acides aminés chargés, la charge net de la séquence et sa composition en chaque élément. Tandis que l'analyse par fenêtre permet de calculer l'hydrophobicité moyenne de chaque fenêtre. Cette procédure fait donc appel au module ***analyse_proteine*** et au module ***matplotlib*** si elle doit tracer des graphiques.

La dernière procédure de ce module, (« *resultats_analyse_seq()* ») permet de réaliser l'analyse de la séquence dans son intégralité en interagissant avec l'ensemble des modules précédemment cités. En premier lieu avec le module ***recuperation_sequence_fasta*** (et donc indirectement avec le module ***lire_fasta***), ce qui permet une première interaction avec l'utilisateur qui peut choisir le type de séquence qu'il souhaite étudier mais aussi s'il souhaite que le programme trace des graphiques ou non. Elle crée ensuite un dossier portant le nom de la description de la séquence dans lequel tous les résultats de l'analyse de cette séquence seront placés. La procédure fait ensuite appel au module ***analyse_proteine*** pour étudier la composition de la séquence (Remarque : la fonction *composition()* permettant cette étude est également présente dans le module ***analyse_ADN*** à l'identique.), si l'utilisateur a demandé à ce que des graphiques soient générés et que le module ***matplotlib*** est présent alors un premier graphique est généré pour représenter la composition de la séquence. La procédure appelle ensuite *resultat_prot()* ou à *resultat_ADN()* selon le type de séquence dont il s'agit. Pour finir l'utilisateur a le choix entre commencer l'analyse d'une nouvelle séquence, approfondir son analyse en la poursuivant sur une ou plusieurs séquences aléatoires de même type et de même

longueur et/ou de même composition, ou bien arrêter le programme. Si l'utilisateur fait le choix d'approfondir son étude la procédure fait alors appel aux fonctions du module ***creation_seq_aleatoire***.

- **Remarques et améliorations possibles**

1. Remarques concernant la structure des modules

Chaque module est composé d'une partie principale appelée « Main » qui permet de tester l'ensemble des fonctions le composant en les lançant automatiquement au moins sur un exemple (parfois plus pour illustrer les différents cas que gère la fonction). Au sein de chaque module chaque fonction est dotée d'une documentation pouvant être appelée par la commande : « nom_de_la_fonction_ou_de_la_procedure**.__doc__ »**.

La stratégie globale adoptée lors de la création de ces modules a été de rendre chaque fonction le plus adaptable possible, tout en évitant les actions ralentissant l'exécution (comme les « in range(…) », les concaténation de liste grâce au signe +, …), en les remplaçant par des équivalents plus rapides (comme « in » ou « in enumerate(…) », « .append(…) », …). Mais aussi en évitant de recalculer les mêmes éléments plusieurs fois grâce à utilisation des arguments par défaut (arguments ayant une valeur par défaut qui entraine le calcul de certaines variables sauf si elles sont données en argument) et en créant des fonctions supplémentaires pouvant calculer plusieurs valeurs simultanément. Une réflexion sur la mémoire a aussi été faite, notamment dans les procédures du module ***analyse_sequence_fasta***. En effet il aurait sûrement été plus lisible de séparer les parties du programme permettant la création de graphique. Cependant, cela aurait impliqué de stocker les valeurs calculées dans les procédures actuelles pour les donner en argument à cette nouvelle procédure, et donc occuper une place plus importante dans la mémoire. Une autre solution aurait été de faire en sorte que cette procédure lise le fichier tabulé qui venait d'être créé et y fasse les transformations inverses de celle qui venaient d'être opérés sur chaque ligne, ou encore de recalculer l'ensembles des valeurs établies dans les autres procédures ce qui aurait représenté une perte de temps et de mémoire considérable. C'est pourquoi le compromis présenté a été choisi.

2. Choix effectués concernant le format des donnée et les critères d'analyse

Au cours de la création de ce programme de nombreux choix ont dû être effectués, notamment sur les critères biologiques permettant l'analyse des séquences, mais aussi sur le format des données récupérables. Ici le choix a été de récupérer uniquement des séquences au format fasta, pouvant se trouver soit dans des fichiers placés dans le répertoire courant soit en ligne. A propos des choix de critères biologiques, plusieurs options étaient envisageables concernant l'hydrophobicité et la charge des acides aminés. En effet il existe plusieurs échelles d'hydrophobicité, ici l'échelle retenue est l'échelle d'hydrophobicité de *Fauchere et Pliska* suggérée dans l'énoncé du projet. Quant aux charges des acides aminés, elles ont été récupérées à pH physiologique (c'est-à-dire 7,4) sur le site « http://protcalc.sourceforge.net » permettant de calculer la charge d'un acide aminé à pH donné.

3. Limites du programme et améliorations envisagées

Le programme est limité dans les possibilités qu'il offre, en effet, comme évoqué précédemment la charge de chaque acide aminé est calculée à pH physiologique uniquement. Une amélioration possible serait donc de permettre un choix entre plusieurs pH. De même une amélioration possible serait de laisser le choix à l'utilisateur entre plusieurs échelles d'hydrophobicité, plutôt que d'imposer l'échelle d'hydrophobicité de *Fauchere et Pliska*.

Une autre limite du programme est qu'il ne permet pas de tracer le graphique représentant les rapport CpG, s'il vaut « NA » pour au moins l'une des fenêtres glissantes. En effet je ne suis pas parvenue à faire tracer par le programme une courbe interrompue dans ce genre de cas, c'est pourquoi j'ai fait le choix de n'afficher que le graphique représentant le pourcentage de cytosine et guanine présent dans chaque fenêtre.

Le format lu est également très limitant, puisqu'il se borne aux fichiers fasta, il serait intéressant de permettre la lecture de format pdb pour les protéines par exemple.

De plus il est important de noter qu'en théorie ce programme fonctionne sur des postes équipés du système d'exploitation Windows, OS X ou Linux. Cependant les tests sur les systèmes d'exploitation autre qu'OS X ont été limités il est donc possible qu'un incident se produise lors de l'utilisation sur des machines équipées de ces autres systèmes d'exploitation.

Intérêt biologique et résultats

- **Intérêt biologique**

Ce programme permet l'étude automatisée relativement approfondie d'une séquence donnée, ainsi que la récupération de l'intégralité des résultats de l'étude par le biologiste. Ainsi l'utilisateur peut rapidement avoir une idée des propriétés de la séquence qu'il a choisi d'étudier, vérifier manuellement les résultats obtenus s'il le souhaite et les interpréter. En effet la connaissance de la position d'éventuels îlots CpG peut permettre d'envisager un rôle de promoteur de gène de vertébré. De même les informations concernant l'hydrophobicité d'une protéine peuvent apporter des indices quant à sa localisation au sein de l'organisme.

De plus la possibilité offerte par le programme d'étudier une ou plusieurs séquences aléatoires de même longueur et/ou de même composition que la séquence de référence peut être très utile pour interpréter correctement les résultats obtenus. En effet, cela permet de vérifier que les données établies ne peuvent pas être le fruit d'un simple hasard mais sont bien le résultat d'une pression de sélection sur une partie de la séquence possiblement lié à son rôle au sein de l'organisme dont elle est issue.

- **Présentation de quelques exemples de résultats**

On s'intéresse ici à trois exemples de séquences nucléiques et trois exemples de séquences protéiques. Quel que soit l'exemple des informations essentielles pouvant aider à l'interprétation se trouve d'ores et déjà dans la description de la séquence fasta. Nous nous

intéresserons dans un premier temps aux trois exemples de séquence nucléiques puis, dans un second temps, aux trois exemples de séquence protéiques.

Lors de l'analyse des séquences nucléiques nous nous intéresserons particulièrement à la présence d'îlots CpG comme évoqué précédemment. Ces îlots sont les représentant de zones présentant un rapport CpG plus élevé que la moyenne (0,6) et un pourcentage de cytosine plus de guanine supérieur à 50%. La présence d'îlots CpG est souvent considéré comme indicatrice de séquence promotrice de gêne chez les vertébrés.

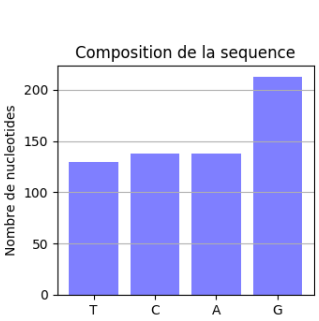
Quant à l'analyse protéique elle portera plus particulièrement sur l'hydrophobicité de la séquence. L'hydrophobicité étant une mesure de l'affinité de la protéine pour l'eau. Plus une protéine est hydrophobe plus elle est repoussée par l'eau et donc est instable dans les milieux aqueux. A l'inverse plus une protéine est hydrophile plus elle est stable dans les milieux aqueux. En effet ces informations peuvent apporter des indices concernant la localisation de la protéine au sein de l'organisme et donc sur le type de rôle qu'elle peut avoir.

De plus nous chercherons à valider les résultats obtenus par des recherches sur des bases de données telles que le NCBI, et si possible à en apprendre un peu plus sur le rôle de la séquence au sein de l'organisme duquel elle est extraite. En effet nous étudions ici des séquences déjà annotées ce qui nous permet de vérifier nos résultats et d'approfondir nos connaissances sur les séquences étudiées pour mieux interpréter ces résultats.

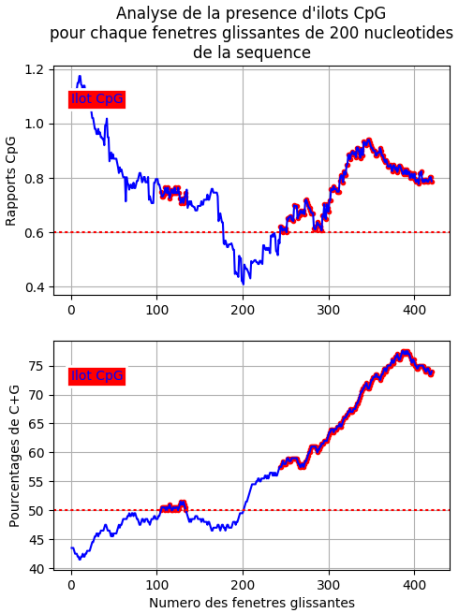
L'analyse sera faite en détail pour une séquence de chaque type, puis pour les séquences restantes plus succinctement.

1. Trois exemples d'analyse de séquences nucléiques

Dans un premier temps nous nous intéresserons à la séquence **FJ767773**. La description de la séquence nous permet déjà de savoir qu'il s'agit d'une séquence du génome humain.



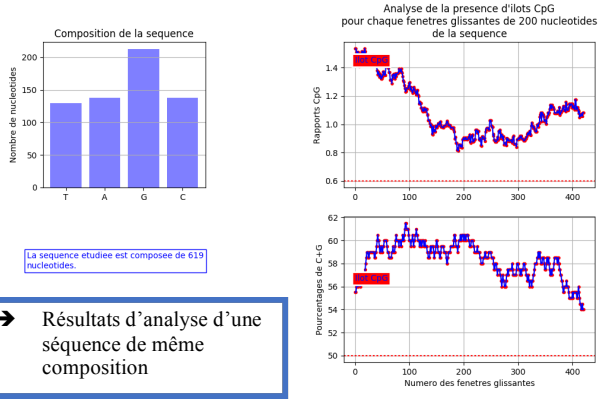
La sequence etudiee est composee de 619 nucleotides.



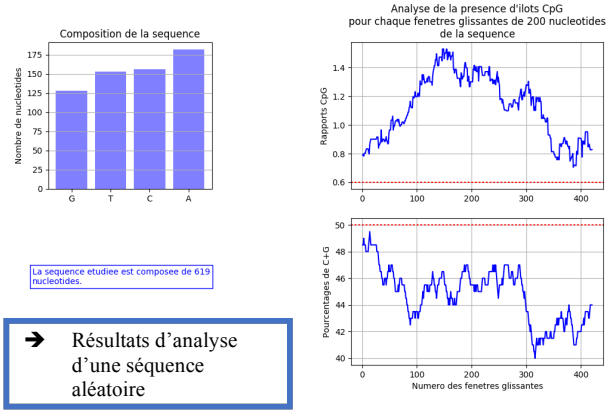
➔ Résultats graphiques de l'analyse de la séquence FJ767773

D'après les résultats graphiques générés par le programme, on peut constater la présence de nombreux îlots CpG relativement concentrés en fin de séquence. Au vu de ces résultats il est

intéressant de compléter l'analyse avec l'étude d'au moins une séquence aléatoire de même longueur et une autre de même composition de cette façon on pourrait s'assurer que ce résultat a peu de chance d'être le fruit du hasard.



➔ Résultats d'analyse d'une séquence de même composition



➔ Résultats d'analyse d'une séquence aléatoire

Au vu de ces analyse complémentaires il semble évident que les observations faites précédemment ne résultent pas d'un hasard. En effet lorsqu'on opère la même étude sur une séquence de même composition que la séquence de référence on observe des îlots CpG éparpillés sur toute la longueur de la séquence. On peut interpréter cela comme le fait que la portion de séquence étudiée est très riche en C et G ce qui par le seul fait du hasard peut engendrer un rapport CpG et un fort pourcentage en C+G tout au long de la séquence. A l'inverse on remarque que l'étude d'une séquence aléatoire de même longueur révèle un rapport CpG élevé mais un pourcentage de C plus G faible ce qui aboutit à une absence total d'îlots CpG. Le résultat concernant le rapport CpG plus élevé de façon aléatoire est cohérent. En effet il a été observé que chez les vertébrés la présence de CpG était sensiblement plus faible que celle qu'on s'attendait statistiquement à trouver. L'explication de cette anomalie proposée en 1967 est une méthylation de la cytosine tendant à convertir progressivement les CpG en TpG.

En outre au vu de la description de la séquence on : « ENA|FJ767773|FJ767773.1 Homo sapiens AMACR gene, promoter region. » on peut corroborer que nos résultats d'analyse ne sont pas aberrants. Cependant pour compléter notre étude de cette séquence et confirmer nos résultats nous avons mené des recherches sur la base de données du NCBI grâce à un BLAST nucléique. Les résultats du BLAST sont très bons puisqu'on trouve de nombreux hits rouges dont le premier à une correspondance de 100% pour une couverture totale de la « Query » (à savoir la séquence de FJ767773). En cliquant sur le numéro d'accession de ce hit on tombe sur une fiche informative de la séquence « Subject » (séquence ayant le meilleur score pour notre recherche, ici l'identité étant de 100% il s'agit de la séquence FJ767773 elle même).

Sur cette fiche informative on retrouve la longueur de la séquence (619 nucléotides), la description de la séquence, l'organisme duquel elle est extraite (l'homme), le gène sur lequel elle se trouve (AMACR), et la confirmation de son rôle : séquence promotrice de la classe régulatrice. Mais on trouve également la position des îlots CpG : de 373 à 533.

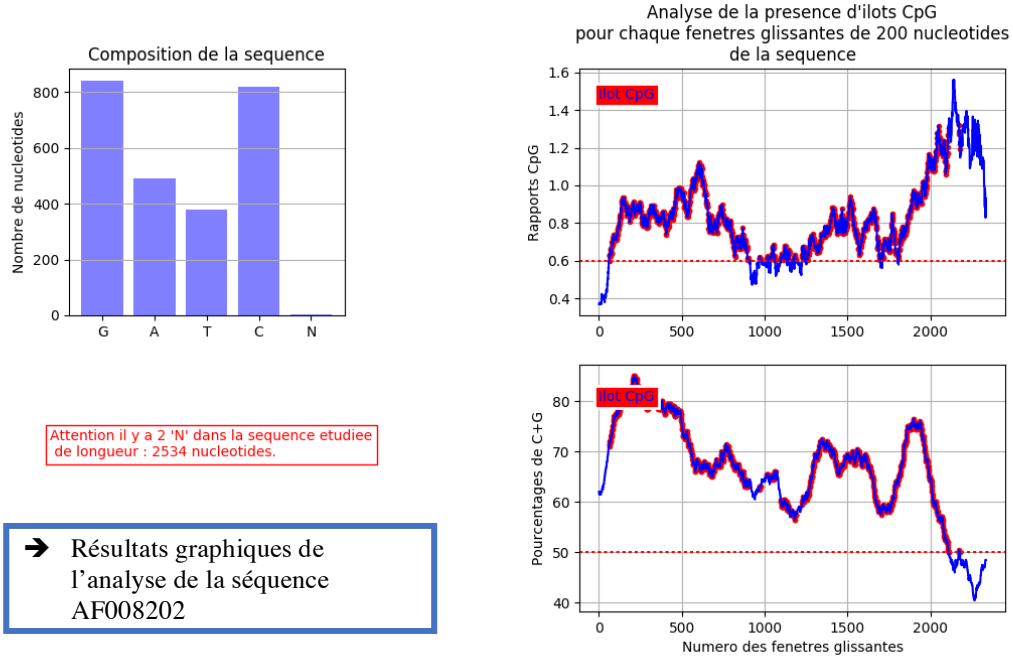
```
  1 ttcctagtgt agtctaaact tttttaaaaa gacatgtaat ccgcggagtt tgtaactcaa
 61 aacgagtgca tctaggaggt atcgcaagcc gtttctggat taaattccca gctagctagc
121 tagctaagca ggggcggtga agaagacaat ctgcagccta gggaagaaaa cgctttcgca
181 ttgttcttac gtgtttacgt tattttattt ccctagagca aggcaggagt agggactcga
241 atggtacagt tgggctgggg atcgccctgg tacaataaaa gcgtccagag aggacggtaa
301 caggcaggag ctccaaaggt cagtccctgc aatttaagac tcaggaattt aggttgcaca
361 aaaaggaaag cacctactg catttggcac tggcggtccc gggaggccgg gggtggggaa
421 gcgcccagtg cgcagactcc gcgggcttgc gcaggcccgc aaaagaggga cgggggtggt
481 gcctttgggg cgcggcgccg cgqctqqqqq cqtqqcqccq cqqctqqqqq cqtqqcqccq
541 gggattggga gggcttcttg ccggctgctg ggctggggct aagggctgct cagtttcctt
601 cagcggggca ctgggaagc
```

D'après ce résultat on peut conclure que le premier groupe d'îlots établi par notre programme n'est pas pris en compte dans ces données. Cependant on peut assurément conclure qu'il s'agit d'une séquence promotrice de gènes chez l'homme. De plus cette fiche permet de connaitre la position du gène sur la séquence, on observe ici que le gêne recouvre toute notre séquence.

En cliquant sur le lien PubMed proposé dans la fiche, on obtient des informations supplémentaires sur le gène dont la séquence étudiée est promotrice (AMACR). On apprend que ce gène joue probablement un rôle dans le cancer du côlon en plus d'avoir un rôle dans le cancer de la prostate (dans les deux cas le gêne est surexprimé). Il est notamment utilisé comme marqueur pour diagnostiquer ce type de cancer.

Pour conclure, la séquence FJ767773 est une séquence promotrice du gêne AMACR qui code pour une enzyme : alpha- methylacyl-CoA racemase impliquer dans la biosynthèse de la bile et la bêta-oxydation des acides gras à chaînes ramifiées. Il s'agit d'un gène essentiel au métabolisme des lipides qui est exprimé dans le foie, le rein, et la vésicule biliaire dans un corps sain.

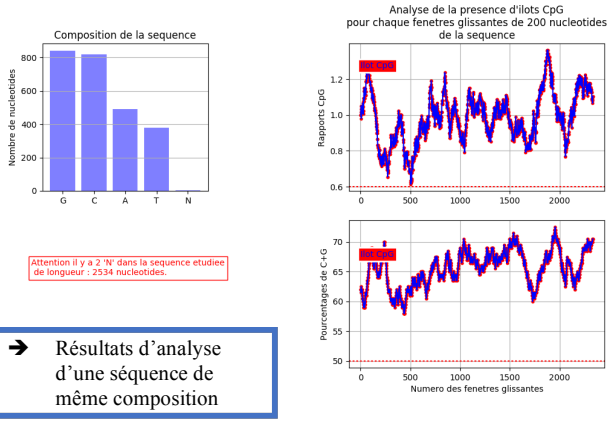
Intéressons-nous à présent plus succinctement à la séquence nucléique **AF008202**, qui d'après sa description provient également du génome humain.



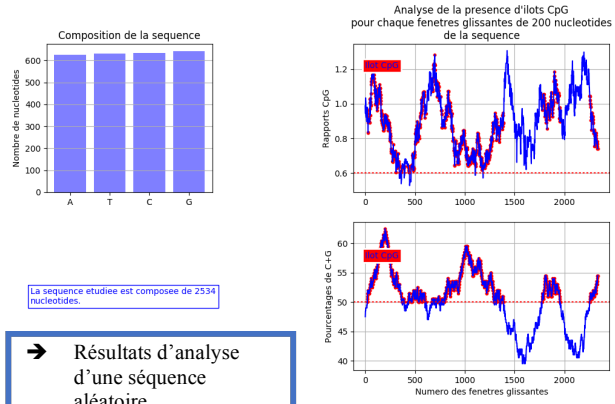
➔ Résultats graphiques de l'analyse de la séquence AF008202

On observe ici de nouveau la présence de nombreux îlots CpG mais répartis sur une plus grande partie de la séquence. Il est également important de noter la présence de deux « N » dans la séquence indiquant un manque d'information sur deux des nucléotides la composant. Cependant, étant donnée la taille de la séquence, à savoir 2534 nucléotides, on peut considérer que ce manque d'information est très ponctuel et n'affectera pas les résultats au point qu'on ne puisse plus les interpréter. Nous poursuivrons donc notre étude sans tenir compte de cette

imprécision. Au vu des résultats il serait intéressant d'approfondir l'analyse en réitérant l'études sur deux séquences aléatoires comme dans l'analyse de la séquence précédente.



➔ Résultats d'analyse d'une séquence de même composition



➔ Résultats d'analyse d'une séquence aléatoire

Contrairement à ce qu'on a observé lors de l'étude précédente, les analyses complémentaires n'apportent pas de résultats très clairs permettant de trancher. En effet quel que soit le type de séquences aléatoires on a l'impression que les résultats observés lors de l'étude de la séquence AF008202 pourraient aussi bien être dû au hasard. Avec ces seules informations il semble donc difficile de trancher.

De plus, la description de la séquence : « ENA|AF008202|AF008202.1 Homo sapiens homeobox protein alx3 (Alx3) gene, CpG island and partial cds. » n'est pas très concluante sur cet aspect. Nous allons donc nous intéresser aux informations présentes sur la base de données du NCBI.

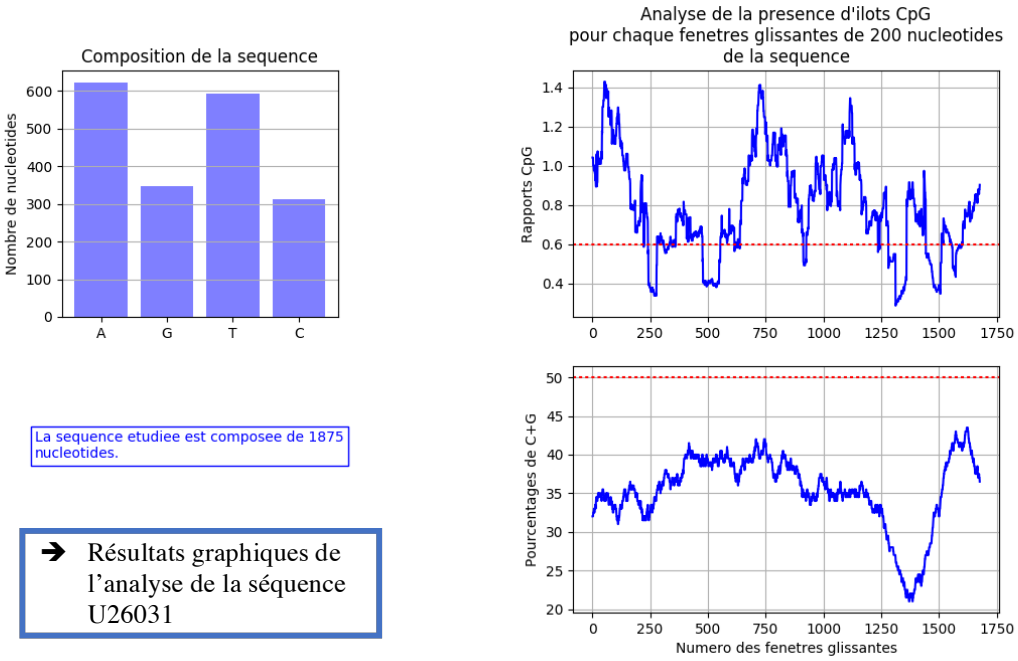
En suivant le même cheminement que précédemment on trouve un hit excellent qui s'avère apparemment être notre séquence. Sur la fiche NCBI correspondante on retrouve à nouveau la position des îlots CpG qui semble globalement correspondre à nos résultats bien que la fin de la séquence ne contienne pas d'îlots CpG dans nos résultats. Les CpG se trouvent de la position 144 à la position 2534.

```
   1 gatatcagga ccacgggtgc tgctctgagc aagagactca cctggatggg tagacaaagg
  61 gatcctgccc tagagaggaa gctgcttcat cctctcctga gctgcctggg actgggctct
 121 gggtgaatgg ggggaggggg ttccggggct ccgacccctt aacccgcacg gggccacacg
 181 tgacgtggat ggttccagca ttaagtcaga ggcgcggccc cctcccctgc cccccgcccc
 241 ccgccccccg cggcgcgcgc cgctcccggg ggctccgcgc ccccgcgccc aggtcctcc
 301 gcttggcagg cgctcacagg cggccgggc agcgcgagcc ccggagcccc ggaggcctgt
                                    ⋮
2401 tttaaagcgt agaatatatc aaaatgagaa caaggggta agaaaattaa cgatttcagt
2461 tgaacggagg acaggccaaa cgaaagggag gcaggcgggt gcgagcagag cctggtagag
2521 atcctggcgg ccgc
```

De plus on observe qu'une partie de la séquence est une CDS (de 376 à 652) comme le laissait présupposer la description. Il s'agit d'une séquence codante pour la protéine homéobox alx3. Ces protéines sont impliquées dans le développement embryonnaire chez les animaux, et ont des rôles de liaison à l'ADN permettant la régulation de l'expression de certains gènes.

Ainsi cette séquence ne semble pas être une séquence promotrice de gène mais a toutefois un rôle indirect dans la régulation de gènes par le biais de la protéine homeobox alx3 pour laquelle elle code.

Prenons désormais comme dernière exemple la séquence **U26031**. Il s'agit cette fois-ci d'un gène de levure.



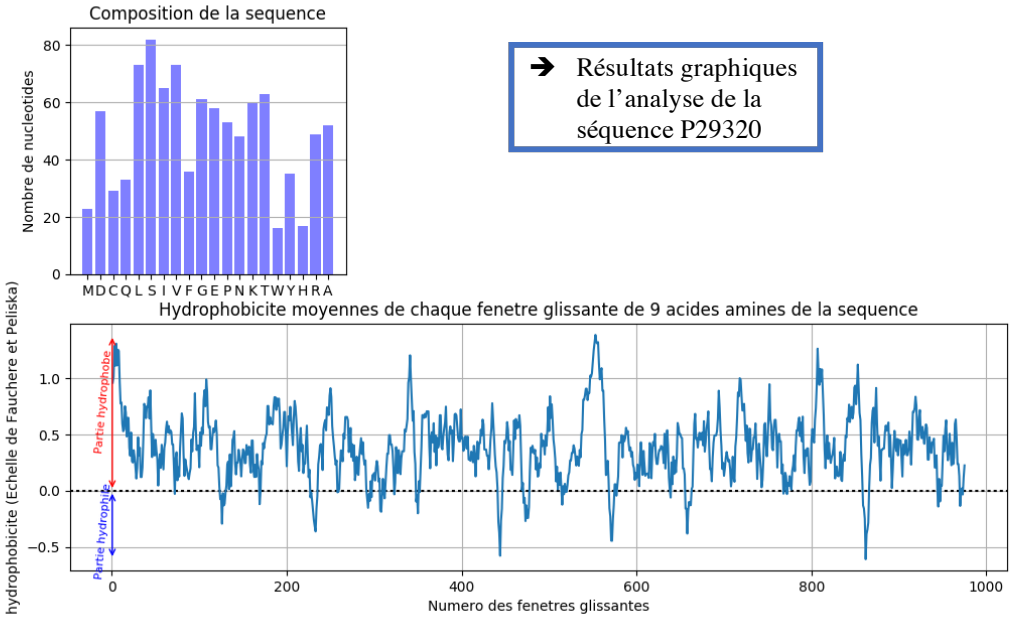
➔ Résultats graphiques de l'analyse de la séquence U26031

On observe ici aucun îlot CpG dans la séquence. Un approfondissement des analyses permet de montrer qu'une séquence aléatoire de même longueur contiendrait des îlots CpG contrairement à une séquence aléatoire de même composition. On peut donc en conclure que cette zone est spécialement pauvre en îlots CpG. De plus la description de la séquence : « ENA|U26031|U26031.1 Saccharomyces cerevisiae replication factor C subunit 5 (RFC5) gene, complete cds. » nous informe qu'il s'agit d'une CDS complète et d'après les recherches effectuées sur la base de données du NCBI, elle se situerait sur la chaine Y12 du chromosome 12 du Saccharomyces cerevisiae.

On peut noter que l'organisme n'étant pas un vertébré une recherche d'îlots CpG n'est pas très pertinente, c'est pourquoi nous n'irons pas plus loin dans l'étude de cette séquence.

2. Trois exemples d'analyse de séquences protéiques

Nous commencerons par l'étude de la séquence protéique **P29320**. Il s'agit d'une protéine humaine.



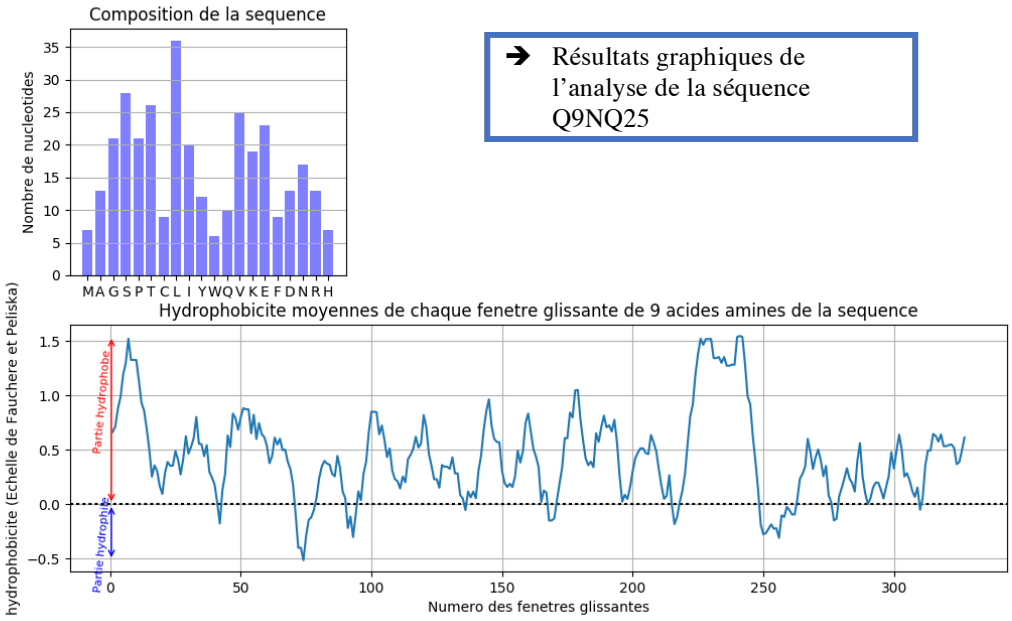
➔ Résultats graphiques de l'analyse de la séquence P29320

D'après ces premiers résultats il semblerait que la protéine soit plutôt hydrophobe. Nous allons donc nous pencher sur la description de la séquence pour essayer d'en apprendre un peu plus : « sp|P29320|EPHA3_HUMAN Ephrin type-A receptor 3 OS=Homo sapiens GN=EPHA3 PE=1 SV=2 ». D'après cette description cette protéine aurait un rôle de récepteur, et au vu de nos résultats on aurait tendance à pencher pour un récepteur membranaire (puisque les membranes constituent des zones hydrophobes) ou du moins transmembranaire puisqu'on observe quelques zones hydrophiles.

Au NCBI de nombreuses informations sont disponibles concernant cette protéine. Pour accéder à ces informations, on fait un BLAST protéique puis on procède de la même façon que pour l'étude de séquence nucléique. On apprend qu'il s'agit d'une protéine de la famille des protéines kinases et plus particulièrement des récepteurs tyrosine kinase qui ont un rôle dans la signalisation intercellulaire. On apprend également que c'est une protéine très exprimée et que son plus haut niveau d'expression a été relevé dans les tissus placentaires. Il est également important de noter que cette fiche nous informe que cette protéine existe sous deux isoformes dont l'un et membranaire tandis que l'autre est sécrétée.

Finalement l'hydrophobicité de cette protéine tiens bien un rôle dans sa fonctionnalité puisque cette protéine présente un isoforme membranaire qui joue un rôle de récepteur dans la communication intercellulaire.

Nous nous intéresserons désormais à la protéine **Q9NQ25**, de nouveau une protéine que l'on trouve chez l'homme.



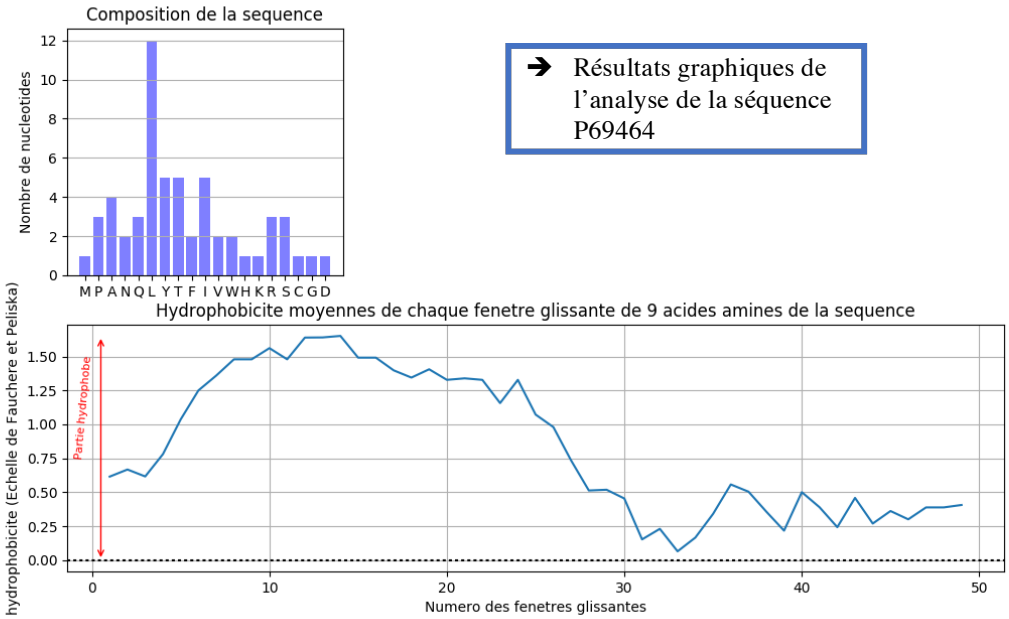
➔ Résultats graphiques de l'analyse de la séquence Q9NQ25

On remarque immédiatement sur ces résultats une forte hétérogénéité de l'hydrophobicité. En effet, certaines zones semblent très hydrophobes tandis que la protéine présente au moins une zone hydrophile en début de séquence. En faisant l'analyse de séquences aléatoires on réalise qu'il est hautement improbable de trouver une zone hydrophobe aussi marquer par simple hasard. On peut donc faire l'hypothèse que cette protéine est localisée dans une zone hydrophobe.

On trouve une fois encore de très nombreuses informations concernant cette protéine au NCBI. On y apprend que cette protéine a un rôle dans la différentiation des cellules de l'immunité et donc est impliqué dans la régulation et la liaison entre la réponse immunitaire innée et la réponse immunitaire adaptative. Cette protéine est ainsi exprimée dans tous les tissus liés à l'immunité tel que les ganglions lymphatiques, la moelle osseuse, la rate … De plus on apprend qu'il existe de nombreux isoformes (au moins sept) de cette protéine et que certains sont transmembranaires, ce qui correspond bien à notre hypothèse de départ. Les propriétés hydrophobes de cette protéine ont donc une importance considérable dans sa localisation ce qui explique qu'elles soient conservées.

Finalement la séquence protéique Q9NQ25 est donc une protéine membranaire ou transmembranaire impliquer dans l'immunité.

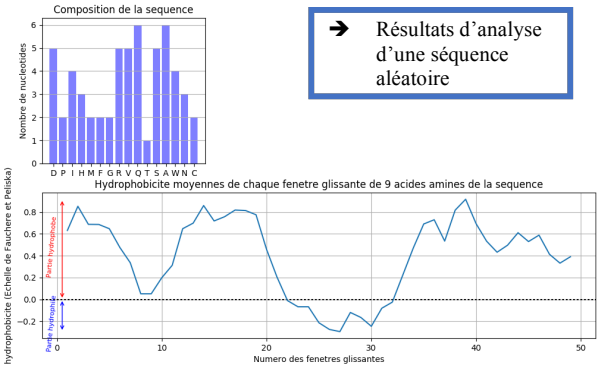
Nous finirons par l'étude de la séquence protéique **P69464**. Il s'agit d'une protéine du virus Mumps.



➔ Résultats graphiques de l'analyse de la séquence P69464

L'analyse par notre programme permet de constater dans un premier temps que cette protéine est entièrement hydrophobe. On cherche donc à approfondir l'étude sur des séquences aléatoire pour se convaincre que cette hydrophobicité à peu de chance d'être due au hasard. Il est donc probable que cette protéine soit membranaire.

➔ Résultats d'analyse d'une séquence de même composition



➔ Résultats d'analyse d'une séquence aléatoire

On observe ici, qu'il ne s'agit en effet pas d'un hasard puisque la séquence aléatoire générée par le programme n'est pas entièrement hydrophobe. C'est donc bien que cette séquence est composée d'un nombre particulièrement élevé d'acides aminés hydrophobes.

D'après sa description et les informations trouvées au NCBI, il s'agit d'une petite protéine de virus qui s'insérerait dans la membrane de l'hôte du virus. En effet la queue des lipides membranaire constituants une zone très hydrophobe on peut trouver un lien entre la nature de la protéine et sa localisation et donc indirectement avec sa fonction.

Conclusion

Finalement ce programme permet une analyse intéressante du point de vu biologique, de séquences nucléiques ou protéique au format fasta, notamment en permettant de compléter cette étude par l'étude de séquences aléatoires de même longueur et ou de même composition. Cette analyse à l'avantage d'être relativement rapide et de générer des graphiques rapidement interprétables.

D'autres part les six exemples étudiés reflètent bien la diversité des séquences pour lesquelles ce type d'analyse peut s'avérer utile et ont permis de tester la robustesse et la relative justesse du programme informatique proposé.

Ainsi ce programme (bien que modeste et améliorable de nombreuses manières) peut s'avérer être une aide précieuse lors de l'étude de nouvelles séquences nucléiques ou protéiques.

Bibliographie et Webographie

- **Charge des acides aminés** : http://protcalc.sourceforge.net
- **BLAST au NCBI** : https://blast.ncbi.nlm.nih.gov/Blast.cgi
- **Aide îlots CpG** : http://www.iro.umontreal.ca/~csuros/IFT6299/H2014/content/prez08-CpG.pdf
- **Rôle de l'alpha- methylacyl-CoA racemase** : E. Scarano, M. Iaccarino, P. Grippo et E. Parisi, « The Heterogeneity of Thymine Methyl Group Origin in DNA Pyrimidine Isostichs of Developing Sea Urchin Embryos », *Proceedings of the National Academy of Sciences of the United Sates of America*, vol. 57, nº 5, mai 1967, p. 1394-1400
- **Aide de vocabulaire** : https://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000025990940
- **Vérification de certain résultat du programme** : http://web.expasy.org/cgi-bin/protparam/protparam