TP 14 bis informatique

BCPST 1 2019-2020

Applications des statistiques

Exercice 1. À faire si vous avez fini les autres exercices. On considère une famille de 100 protéines relativement proches dont les séquences alignées sont données dans le fichier "proteines". Chaque lettre représente un acide aminé. Les 20 lettres permettant de représenter ces éléments sont les suivantes : A R N D C Q E G H I L K M F P S T W Y V.

On cherche à déterminer la probabilité de mutations d'un acide aminé en un autre à l'aide des données fournies.

Pour simplifier l'implémentation, on pourra dans la suite utiliser la fonction suivante qui prend en argument un caractère et qui lui associe un nombre compris entre 0 et 19, cette association étant bijective.

```
def chaine_vers_nombre(c) :
    acide_amine = "ARNDCQEGHILKMFPSTWYV"
    for i in rang(len(acide_amine)) :
        if acide_amine[i] == c :
            return i
    return -1 # cas où le caractère ne représente pas un acide aminé.
```

Ainsi, A et 0 sont en correspondance, R et 1 sont en correspondance etc.

1. Récupérer le fichier "protéines". Pour stocker les chaînes de caractères dans une liste, exécuter les commandes suivantes :
   — fichier=open("nom complet du fichier","r")
   — L=fichier.read().split("\ n")
   — del(L[-1])
   L contiendra alors une liste de 100 chaînes de caractères.
2. Écrire une fonction qui prend en argument une chaîne de caractères et qui retourne la liste L des nombres d'occurrences des acides aminés. On respectera l'ordre donné pour les acides aminés. Ainsi, $L[0]$ est le nombre de A, $L[1]$ le nombre de R etc.
   En déduire le nombre d'occurrences de chacun des acides aminés présents dans le fichier.
   En déduire la fréquence relative de chacun de ces acides aminés. On rappelle que la fréquence relative de i est la quantité $(occ\ i)/(total)$.
3. Écrire une fonction `diff(c1,c2)` qui prend en argument deux chaînes de caractères de même taille (correspondant à des protéines) et qui retourne une matrice M de taille 20×20 telle que :
$$\forall (i,j) \in \{0, \cdots, 19\}^2, m_{ij} = \text{Card}\{k | c1[k] = i, c2[k] = j\}.$$
   (L'indexation commence à 0 car on est en informatique.) Déterminer la matrice correspondante pour les deux premières chaînes caractères du fichier "protéines".
4. En déduire une fonction qui prend en argument une liste de chaînes de caractères L (correspondant à une liste de protéines) et qui retourne une matrice M de taille 20×20 telle que
$$\forall (i,j) \in \{0, \cdots, 19\}^2, m_{ij} = \text{Card}\{(k,m,n) | L[m][k] = i, L[n][k] = j\}.$$
   Déterminer la matrice M correspondante pour les 100 protéines.
5. On cherche à construire la matrice A de taille 20×20 où $A_{i,j}$ est la probabilité que i mute en j dans un temps donné T. Pour cela, on construit d'abord la matrice M de la question précédente. À partir de celle-ci, on en déduit la mutabilité de chaque acide aminé, que l'on définit par
$$\forall i \in \{0, 1, \ldots, 19\}, M_i = \frac{\sum_{0 \leq j \leq 19, j \neq i} m_{ij}}{\sum_{0 \leq j \leq 19} m_{ij}}.$$

Elle correspond à la probabilité que i mute. Le coefficient A_{ij} de A est alors obtenu de la manière suivante

$$\begin{array}{ll} \text{si } i \neq j & A_{ij} = \frac{M_j m_{ij}}{\sum_j m_{ij}} \\ \text{si } i = j & A_{ii} = 1 - M_i \end{array}$$

A_{ij} correspond donc à la probabilité que i mute en j. Construire la matrice A à l'aide de la matrice M précédente.

Remarque 1. Des matrices de mutations on été utilisées pour déterminer des matrices de "score" qui sont à la base de méthodes d'alignement de séquences. La construction de la matrice présentée a été utilisée pour construire les matrices PAM, qui ont été introduites en 1978 par Margaret Dayhoff. Bien qu'il y ait d'autres familles de matrices, celles-ci sont encore utilisées aujourd'hui pour l'alignement des séquences de protéines. On pourra consulter l'article " A model of evolutionary Change in Protein" (M.O Dayhoff, R.M Schwartz, B.C Orcutt) pour plus de détails.