IFT3295 - TP1

18 septembre 2016

Ce TP est à faire en équipe de deux ou seul si vous le préferez. Vous devez le rendre au plus tard le lundi 3 Octobre avant la démo.

Assemblage de séquences (70pts)

Mise en situation

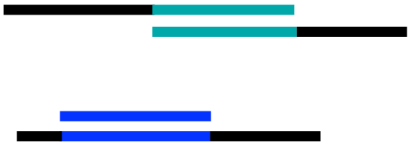
Le laboratoire du Dr Osborn étudie activement la protéine p53, un suppresseur de tumeurs souvent endommagé dans plus de la moitié des cancers humains. p53 est un facteur de transcription qui régule plusieurs fonctions cellulaires importantes comme la mitose et l'apoptose cellulaire. Au cours d'une expérience de co-immunoprécipitation, le Dr Osborn a identifié une-protéine X interragissant avec p53. Il décide donc d'étudier cette nouvelle protéine qui constituer une piste pour la découverte d'un traitement contre le cancer. Après séquençage, Dr Osborn obtient une série de fragment de séquences nucléotidiques qu'il faut ensuite assembler pour obtenir la séquence du gène X. Il requiert votre assistance pour cette tâche.

Chevauchement de séquences (25pts)

Soit deux séquences $Y = ACGCATT$ et $X = GATACGTCACGTGCACGG$
Pondération :

- "match" : +1
- "mismatch" : -1
- "indel" (insertion ou suppression) : -2

On veut un alignement entre X et Y tel que le chevauchement entre ces deux séquences soit maximal (comme illustré sur la figure suivante).



1. En quoi un tel alignement est-il différent d'un alignement local ?
2. Quelles doivent être les valeurs de la première ligne ($V(0, j) \quad \forall j$) ? et celles de la première colonne ($V(i, 0) \quad \forall i$) de la table de programmation dynamique V ? Justifiez votre réponse.
3. Donnez les équations de récurrences à utiliser pour remplir la table de programmation dynamique.
4. Déduisez-en un algorithme pour trouver **l'alignement** maximisant le chevauchement entre deux séquences et son **score** en fonction des coûts donnés plus haut. Vous devez remettre un code qui retourne pour une paire de séquence le score ainsi qu'un alignement correspondant.
5. Donnez la table de programmation dynamique pour le chevauchement entre X et Y, ainsi que l'alignement obtenu et sont score.
6. Expliquez pourquoi l'utilisation de scores positifs pour les mismatch (par exemple 1) est une mauvaise idée pour ce problème.

Assemblage de fragments (20pts)

1. À partir des fragments suivants :
   - S_1 : $AACTCTCTACTGCTTTCCCC$
   - S_2 : $CTACTGCTTTCCCCGCCGGA$
   - S_3 : $CTTTCCCCGCCGGAACCTTCAC$
   - S_4 : $TAAATTACAACTCTCTACTA$

   obtenus par le Dr Osborn et de votre algorithme d'alignement, proposez une séquence assemblée plausible (**contig**) qui montre la position de chaque fragment.
2. Quelle est la longueur de votre contig final ?

Traduction et Recherche Blast (25pts)

Grâce à votre aide, le Dr Osborn a finalement obtenu la séquence de l'ARNm du gène X (*geneX.fasta*). Il désire maintenant identifier ce gène.

1. Trouvez la séquence de la protéine produite par le gène X. Pour ce faite traduisez d'abord la séquence nucléotidique dans les différents cadres de lecture, en considérant qu'il s'agit du brin codant [1]. Utilisez le code génétique standard (disponible ici ==> table génétique). Veuillez noter que les nucléotides et les acides aminés sont toujours représentés par un seul caractère dans les séquences et que le signal de terminaison de la traduction (stop) n'est pas ajouté dans la séquence protéique.
2. Quelle est la longueur (en acide aminés) de votre protéine traduite X ?
3. En vous servant de l'outil Blastp et/ou de uniprot , identifiez :
   - le nom de la protéine X,
   - l'espèce d'origine,
   - sa fonction,
   - sa position génomique

Autre variante du problème d'alignement de deux séquences (30pts)

Nous nous intéressons ici à une variante du problème d'alignement entre deux séquences X et Y. On désire obtenir un alignement entre un facteur de X' de X et Y, tel que le score de cet alignement soit le meilleur parmi tous les facteurs de X [2]. Un alignement global de X avec Y ne fonctionnerait pas, puisqu'on alignerait X en entier avec Y. Un alignement local ne fonctionnerait pas non plus, puisque Y ne serait pas nécessairement aligné en entier avec X'.

Vous devez remplir une table D à l'aide d'un algorithme de programmation dynamique basé sur les algorithmes d'alignement vu en cours. Pour ce faire, répondez aux questions suivantes :

1. Que représente la valeur de la case $D(i, j)$

1. l'orf le plus long encode souvent la protéine d'intérêt
2. X' est un facteur de $X = x_1x_2...x_n$ si $X = x_ix_{i+1}..x_j$ tel que $1 \leq i \leq j \leq n$ ou X' est vide.

2. Comment initialiserez vous la première ligne $D(0, j)$ et la première colonne $D(i, 0)$?
3. Quelle(s) cellule(s) de D doit on regarder pour retrouver le score maximal cherché ?
4. Comment retrouver un facteur optimal de X et son alignement avec Y ?
5. Remplissez la table pour les séquences X et Y ainsi que les scores de la section précédente.
6. Comparer la complexité de votre algo à celle de l'algorithme naif qui consiste à énumérer tous les facteurs de X, puis à les aligner avec Y afin de déterminer le facteur avec le meilleur score d'alignement.