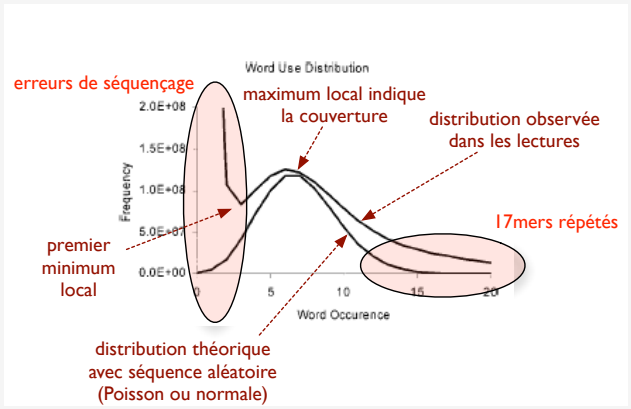
3. TP: correction d'erreurs dans lectures de séquencage

3.0 Introduction au projet

Le but de ce TP est d'explorer les méthodes de correction d'erreur de séquençage à l'aide de fréquence de mots dans les lectures. Le principe de la correction est qu'une erreur de séquençage crée des k-mers rares si $4^k \gg L$, où L est la longueur du génome séquencé. En conséquence, on peut corriger l'erreur en cherchant des positions où la modification d'une nucléotide rend les k-mers fréquents.



Afin de fixer les notions de «rare» et «fréquent», on doit examiner la distribution de nombre d'occurrences. On définit un *mot* comme un k-mer avec égalité à son complément inverse. (Donc `ACTTG` = `CAAGT`.) Dans cet exercice, le *spectrum* est la distribution de fréquence de mots : $f(0)$ est le nombre de mots qui n'occurrent jamais, $f(1)$ est le nombre de mots qui se trouvent une fois parmi toutes les lectures, etc.

Mullikin & Ning. *Genome Research* 13 :81–90 (2003)

3.1 Modèle théorique et algorithme (20 points)

a. Nombre de mots (2 points) ▶ Quel est le nombre $N(k)$ de mots différents en fonction de la longueur de mots k ? (Considérez égalité avec le complément inverse ; notez que la formule n'est pas la même pour k paire et impaire.) Exemples : $N(1) = 2, N(2) = 8$.

b. Approximation Poisson (4 points) Dans le modèle théorique usuel, on assume que la fréquence d'un mot qui apparaît une fois dans le génome suit la distribution Poisson avec espérance c parmi les lectures où c est la couverture. ▶ Donnez une formule pour la valeur $f(i)$ en espérance dans le modèle Poisson, en supposant qu'un mot n'apparaît qu'une fois (ou jamais) dans le génome de longueur L.

c. Spectrum observé (2 points) Supposons qu'on a calculé $f(i) : \ = 0, 1, 2, \ldots$ à partir d'un ensemble de lectures. ▶ Donnez un algorithme qui identifie les plus petits extrema de la fonction. En particulier, on veut i_0, la fréquence i où $f(i)$ atteint son premier minimum, et $i_{\max}$ où $f(i)$ atteint son premier maximum après i_0.

d. Calcul du spectrum (6 points) Implantez un logiciel pour calculer le spectrum. Le programme prend un fichier FASTQ et la longueur k à l'entrée, et doit afficher $f(0), f(1), \ldots$, ainsi que les extrema $i_0, i_{\max}$ de l'exercice **c.** (Il suffit juste deux colonnes séparées par TAB : i et $f(i)$ pour tout i avec $f(i) > 0$, et une dernière ligne avec i_0 et $i_{\max}$.) Vous pouvez supposer que $k \leq 15$. Testez votre programme avec les lectures du projet de séquençage de levures [Bergström et al. (2014) "A high-definition view of functional genetic variation from natural yeast genomes" *Molecular Biology and Evolution*, DOI :10.1093/molbev/msu037] : http://www.moseslab.csb.utoronto.ca/sgrp/download.html. Utilisez $k = 13$ dans les tests.

e. Algorithme de correction (6 points) On a une lecture $s[1..\ell]$ avec les mots $w_1 = s[1..k]$, $w_2 = s[2..k+1]$, …, $w_{\ell-k+1} = s[\ell-k+1..\ell]$, et les valeurs de qualité correspondantes $q[1..\ell]$. Soit $\mathsf{occ}[w]$ le nombre d'occurrences du mot w dans l'ensemble de lectures. Un mot est *rare* si $\mathsf{occ}[w] \leq i_0$. ▶ Proposez un algorithme qui «corrige» la lecture à une seule position. L'algorithme prend s, q et i_0 à l'entrée, et il a accès à occ. Si aucun mot n'est rare, l'algorithme n'a rien à faire. Sinon, l'algorithme doit identifier une seule modification $s[j] : b \to b'$. (C'est à vous à décider comment cette position est choisie — en considérant les valeurs de qualité ou non.) Donnez le temps de calcul en fonction de ℓ et k.

3.2 Logistique

Remise. Remettez un rapport en PDF (a–e), et le code source (d) par e-mail à csuros@iro.umontreal. . . .

Date limite. Soumettez avant 21 :00 le 27 avril.