Projet d'Algorithmique pour la Génomique

Développement d'une solution de mapping de données de séquençage à haut-débit sur un génome de référence

Nicolas Parisot & Sergio Peignier

(D'après Alban Mancheron et Vincent Ranwez)

1. Objectif

L'objectif de ce projet est de développer une stratégie de « mapping » de séquences génomiques courtes, appelées "*reads*", sur un génome de référence donné. Les reads sont obtenus par une technique de séquençage haut-débit (*e.g.* Illumina). Ainsi, ces *reads* peuvent comporter des erreurs de séquençage ou des variations biologiques naturelles (*i.e.* SNP, indels, translocations etc.) par rapport au génome de référence.

La méthode que nous vous proposons d'implémenter consiste à considérer les placements exacts des sous-mots de longueur `k` (appelé `k`-mers) de chaque read sur le génome complet et de définir des critères algorithmiques permettant de localiser la (ou les) occurrence(s) des reads sur le génome, en analysant les différentes situations pouvant se produire :

- le cas le plus trivial correspond à la situation où tous les `k`-mers du read analysé sont localisés de façon unique, dans le bon ordre et sur le même brin du génome, alors le read a une occurrence unique exacte à la position définie par l'occurrence du premier `k`-mers du read sur le génome. Cette situation est a priori très rare, bien évidemment.

- une première variante de ce cas correspond à la situation où les reads sont localisés sur le brin complémentaire inversé.

- un cas relativement trivial est similaire aux deux premiers, à ceci près que certains `k`-mers ont plusieurs occurrences, mais qu'il n'existe qu'une seule suite ordonnée valide de positions des `k`-mers. Cette situation est a priori plus réaliste que la précédente mais demeure un cas encore trop particulier.

- un cas plus fréquent correspond à la présence d'une petite mutation ou d'une erreur de séquençage dans le read. Deux cas de figure se présentent alors :
  - supposons que cette mutation ou erreur survienne vers le milieu du read. Alors les premiers et derniers `k`-mers du read peuvent permettre d'une part de positionner correctement le read sur le génome, mais également de diagnostiquer la variation

observée. En effet, si les `k`-mers du milieu (qui ne sont pas positionnés « au bon endroit », voire pas positionnés du tout, sur le génome de référence) ont un nombre d'occurrences similaire aux `k`-mers correctement positionnés, alors la variation entre le read et le génome est probablement d'origine biologique. Au contraire, si les `k`-mers du milieu ont un nombre d'occurrences très en dessous de celui des `k`-mers correctement positionnés, alors il s'agit vraisemblablement d'une erreur de séquençage. Dans tous les cas, le nombre de `k`-mers qui ne sont pas correctement positionnés ainsi que le positionnement des `k`-mers aux extrémités permettra de définir, par un simple calcul, l'origine de la variation (insertion, suppression, substitution). Le nombre d'occurrences d'un `k`-mer est appelé son « support ».

- supposons maintenant que cette mutation ou erreur survienne vers une des extrémités du read. Alors selon que le nombre de `k`-mers correctement localisés est significatif ou pas il sera possible ou non de localiser correctement le read. Il sera toujours possible d'identifier s'il s'agit d'une variation d'origine biologique ou bien d'une erreur, cependant il ne sera pas possible de diagnostiquer plus finement la variation.

- d'autres cas de figure sont possibles et correspondent à des situations «topographiques» des localisations des `k`-mers sur le génome de référence. Ces situations sont nombreuses, plus complexes et ne font pas l'objet de ce projet. Cependant rien ne vous interdit d'en évoquer quelques-unes avec les solutions que vous pourriez mettre en place, et il est fortement conseillé de structurer votre code de sorte que ces cas puissent être facilement intégrés dans votre analyse.

2. Mise en œuvre

Pour élaborer votre méthode de mapping, il est nécessaire de pouvoir lire et analyser des fichiers de données de séquençage. La première étape du projet consiste donc à prendre en main la bibliothèque `Biopython` pour manipuler les fichiers au format FASTQ.

Dans un second temps, il vous faudra définir une structure de données permettant de chercher des mots de longueur fixe ($k > 0$) à partir d'un texte `t` de longueur `n` et capable de répondre efficacement aux questions suivantes :

- Étant donné un mot `w` de longueur `k`, est-il présent (lui ou son complémentaire inversé) dans le texte indexé `t` ?
  - le cas échéant, combien de fois apparaît-il (quel est son support), à quelles positions et sur quel brin ?

- Quel est le mot de longueur `k` présent à la position `i` ($0<i\leq n$) dans le texte `t` ?

- Bonus : proposez une méthode de recherche de sous-séquences approchée autorisant l'alignement de sous-séquences avec un nombre maximal d'erreurs `d`.

Dans le choix de la structure, vous veillerez bien à prendre en compte le fait que :

1. le texte `t` est formé sur un petit alphabet (`A, C, G` et `T`)
2. le texte `t` peut-être très grand ($n > 10^9$)
3. une structure de données peut être constituée de plusieurs structures distinctes.

Enfin, vous implémenterez votre solution algorithmique de mapping d'un fichier de reads sur un génome de référence. La sortie de votre algorithme sera au minimum la localisation des reads sur le génome.

3. Évaluation

Vous appliquerez votre algorithme de mapping sur le jeu de données simulé disponible sur Moodle (et le génome de référence correspondant). Un fichier au format BAM contenant les positions attendues des reads est également disponible sur Moodle afin de mesurer les performances de votre méthode.

Vous veillerez à mesurer et à présenter la complexité temporelle de votre algorithme, ainsi que les temps de calculs réels observés lors de vos expérimentations. Réfléchissez également à la pertinence d'utiliser des méthodes de "nettoyage" des jeux de données de séquençage (e.g. Trimmomatic) pour améliorer les performances de mapping.

En trinômes, vous devrez fournir un notebook qui contiendra i) l'algorithme de mapping, sa description et sa complexité, ii) son exécution sur le jeu de données simulé et l'évaluation du temps de calcul, et iii) les résultats obtenus et l'explication du protocole expérimental de comparaison à la vérité terrain (fichier BAM).