25 Décembre 2020

HMIN113M - Rapport projet VCF
Système BCD :
Analyse et visualisation de données VCF

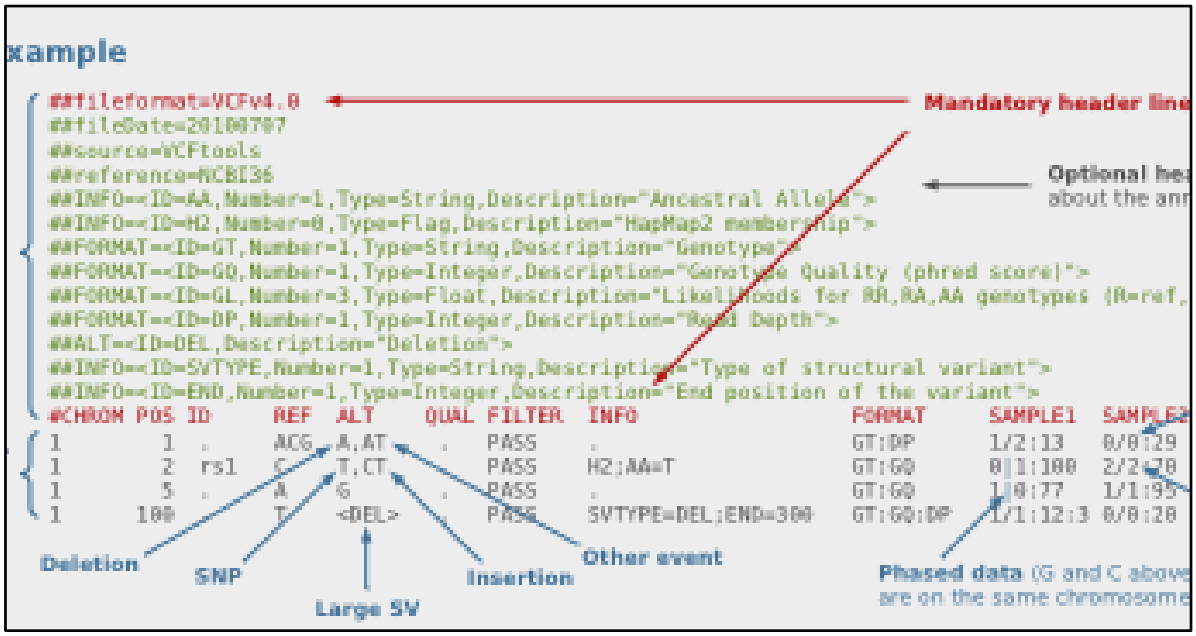

Image descriptif du contenu d'un fichier VCF
Source : https://1.bp.blogspot.com/-iPlBS8NNJtA/UPR4Yg-I2EI/AAAAAAAAERY/qCOttGOt4XI/w1200-h630-p-k-no-nu/vcf.png

Université de Montpellier – Master Sciences et Numériques pour la santé (SNS) parcours Bioinformatique, Connaissances et Données (BCD)

Etudiants

- *IMBERT Jacques*
- *DRAGO Cyril*

Introduction :

Le format VCF (Variant Call Format) est un format de fichier texte contenant les variants génétiques d'un génome ou séquence de référence.
Ce type de fichier montre ainsi les variants structuraux de séquences alignés sur une séquence ou génome de référénce, cela permet notamment de faire du génotypage, c'est-à-dire de relever les différences génétiques entre le génome ou séquence d'individus avec le génome ou la séquence de référence.
Dans le domaine de la santé, cela permet, par exemple, de détecter les variants génétiques potentiellement responsables de certaines maladies, en comparant le génome d'individus malades avec le génome de référence « sain » de l'homme.
Nous avons choisi de faire un projet d'analyse sur des fichiers de format VCF dans le cadre du projet HMIN113M car nous trouvons que ces fichiers renferment beaucoup de données et connaissances sur des variants génétiques, une grande quantité de données plus ou moins annotés dont l'analyse peut apporter beaucoup d'informations et réponses. Nous pourrions par analyse de ces fichiers, par exemple, mieux comprendre les maladies génétiques en détectant les différentes variations génétiques causant la maladie et en étudiant ces variations.

Les variants structuraux détectés par le fichier sont des substitutions nucléotidiques, insertions, délétions ou autre événements entre un génome de référence et des séquences alignés dessus. On peut voir de quoi est composé un fichier VCF sur la figure 1. Celui-ci contient deux grandes parties, une partie « en-tête » et une partie « corps » traduit respectivement par « header » et « body » sur la figure 1.

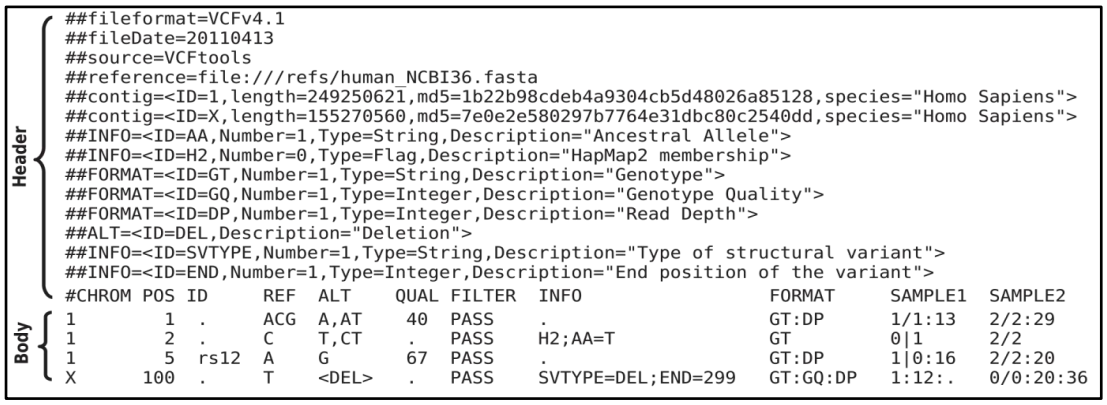

Figure 1 : Exemple type d'un fichier VCF

On retrouve dans la partie « en-tête » des méta-informations sur le fichier. Cette partie commence ses lignes par « ## » et ensuite décrit des informations globales du fichier. Seule la première ligne « ##fileformat==la_version_du_format_VCF » est obligatoire dans cette partie du fichier, le reste sont des options permettant de mieux comprendre les données à analyser. Les méta-informations qui sont très recommandés d'implanter sont les lignes « ##INFO » qui informent sur les variants (le nombre d'échantillons, si c'est un allèle ancestral…), les lignes « ##FILTER » qui renseigne sous quelles contraintes les variants ont été filtrés, et enfin les lignes « ##FORMAT » qui informent sur les échantillons des variants. Il faut savoir que le filtre apposé sur les données relèvent généralement d'un score de qualité « QUAL » qui représente la qualité de l'alignement, plus ce score est élevée moins l'alignement de séquence a de chances d'être une erreur. Par convention on filtre selon un score de qualité supérieur ou égal à 20, ce qui signifie qu'on filtre les variants détectés pour que le risque d'erreur sur l'alignement ne dépasse pas 1%.

Dans la partie « corps », on retrouve les lignes de données pour chaque variant détecté aux différentes positions du génome de référence. Ces lignes forment plusieurs colonnes donnant plusieurs informations sur chaque variant, le titre de l'information rapporté par chaque colonne est noté dans une première ligne obligatoire commençant par un « # ». Nous pouvons observer dans la figure 1, que les informations principales de chaque colonne sont : « CHROM » pour le chromosome sur lequel se

situe le variant, « POS » pour la position nucléotidique du variant sur la séquence, « ID » pour un identificateur de la variation (un « . » signifie que cela est inconnue), « REF » montrant la séquence nucléotidique de référence, « ALT » montrant la ou les séquences qui varient en comparaison avec la séquence de référénce, « QUAL » montrant le score de qualité, « FILTER » marquant « PASS » si le variant a passé le filtre de qualité, « INFO » et « FORMAT » donnent des méta-informations des variants liés à ce qui est marqué en en-tête.

Nous avons donc eu pour projet de créer un script Python3 permettant d'analyser les données de n'importe quel fichier VCF. Ce script devant vérifier que c'est un fichier VCF qui est entré, puis l'ouvrir, stocker les données et informations des variants de l'en-tête et du corps du fichier dans des dictionnaires, et enfin effectuer des analyses sur eux qui devront être affichés à l'utilisateur optionnellement par une interface graphique.
Nous nous sommes donnés ces objectifs, et tenions fortement pour notre propre satisfaction à créer une interface graphique facile d'utilisation, ainsi que des analyses simples à comprendre.

Les accomplissements du projet :

Le programme se déroule en plusieurs étapes :

1) Création de la fenêtre graphique
2) Stockage des données dans des dictionnaires
3) Analyse

L'utilisation du programme est basée sur l'interaction de l'utilisateur avec la fenêtre graphique créée au début. Celle-ci est composée de 3 boutons, et 3 menus déroulants.

Le premier bouton "Sélectionner un fichier" permet de choisir un fichier dans l'ordinateur, sur lequel effectuer les analyses.
Nous effectuons les vérifications suivantes : le fichier doit être un .vfc, et nous devons être capables de l'ouvrir. Si un de ces tests n'est pas passé, le fichier est considéré comme non valide.
Après sélection d'un fichier valide, nous procédons à la lecture de toutes ses lignes, et le stockage de celles-ci dans des dictionnaires. Les lignes sont différenciées par leurs caractères du début :

- si elles commencent par "##" : lignes d'informations
- si elles commencent par "#" : lignes des noms des colonnes
- sinon : lignes des variants

Chaque ligne jugée nécessaire est lue et traitée, puis stockée dans un dictionnaire.
Pour les lignes concernant les balises : nous établissons une norme (une ligne est transformée en un dictionnaire organisé selon le format suivant : {ID : X, Number : Y, Type : Z, Description : W}), et on stocke la ligne dans un dictionnaire. Si la ligne ne respecte pas ce format nous établissons des champs par défaut.
Exemple : « Number : 0 implique Type : FLAG »
A ces informations, nous ajoutons un set de balises connues déjà utilisées par la communauté.

Nous traitons ensuite les variants : pour chaque variant, nous remplissons un dictionnaire de la forme {'CHROM' : [], 'POS' : [], 'ID' : [], 'REF' : [], 'ALT' : []...} que nous ajoutons à une liste. Chaque élément de cette liste représente donc un variant. Il est ensuite aisé d'accéder à ces informations pour pratiquer des analyses.

Une fois un fichier sélectionné, il est possible d'en choisir un autre. Les dictionnaires et listes seront alors vidés entièrement, et remplacés par les nouvelles valeurs. Aucunes informations concernant un fichier précédemment ouvert ne sont conservées.

Les 3 menus déroulants permettent de choisir l'analyse à effectuer, le chromosome sur lequel effectuer cette analyse (ce dernier menu est mis à jour selon le fichier sélectionné), et le score de qualité minimale pour chaque variant. Ce dernier permet à l'utilisateur d'avoir un filtre de qualité qu'il choisit selon ses critères, et est particulièrement utile si le fichier VCF ne contient pas de filtre sur la qualité, car les alignements de mauvaises qualités ne devraient pas être analysés.

Il convient ensuite d'appuyer sur le bouton "Analyse", afin de lancer la fonction d'analyse correspondante. Afin de rendre ce travail plus simple, nous avons créé un dictionnaire avec en clés les noms des analyse du menu déroulant, et en valeur le nom de la fonction à lancer. Nous utilisons donc le nom de l'analyse sélectionné et lançons la fonction d'analyse associée comme montré dans la figure 2.

Il est possible que l'analyse n'ait rien à afficher, dans le cas où les données sont manquantes, une fenêtre d'erreur apparaitra alors.
Le bouton "Analyse Random" suit le même chemin qu'une analyse normale, en choisissant une fonction au hasard dans le dictionnaire.

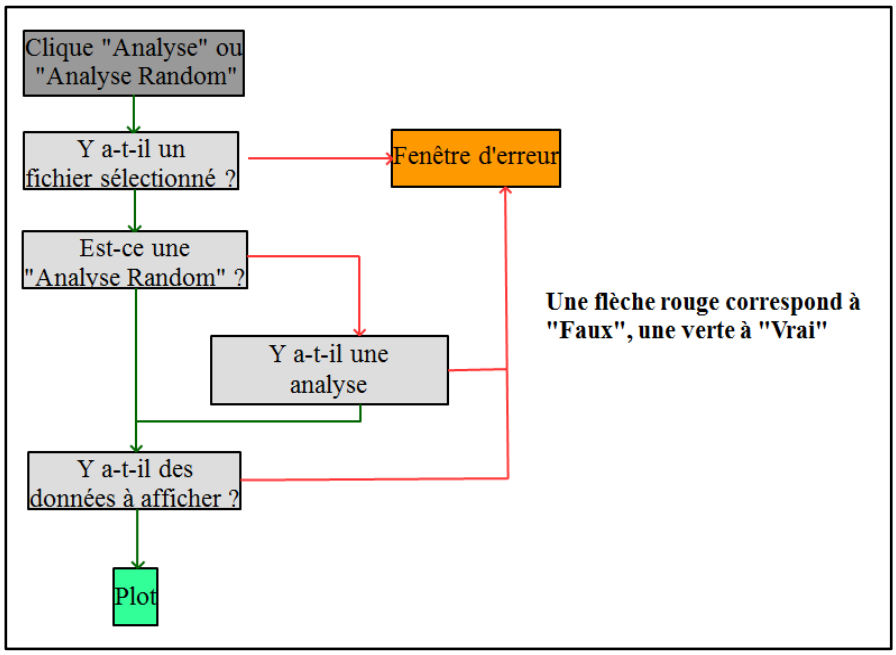

Figure 2 : Schéma du fonctionnement du script lors du clic sur le bouton « Analyse » ou « Analyse random »

Les analyses réalisées :

Nous proposons différentes analyse, celles-ci ne variant pas selon le fichier sélectionné.
Il est possible de se renseigner sur la qualité, le génome de référence, le passage du filtre, et les insertions/délétions qui ont eu lieu.
De plus, une analyse peut être faite sur le total des variants, ou bien sur les variants d'un chromosome en particulier. Cela est utile si l'utilisateur souhaite observer les variants génétiques d'un chromosome en particulier, ou s'il souhaite voir les différences observées lors des analyses entre les différents chromosomes.

Chaque analyse est affichée en une ou plusieurs "pie chart", ou bien des diagrammes, ou des histogrammes. Ces figures sont réalisées à l'aide de la librairie Matplotlib.
Nous pratiquons des analyses statistiques en fonction des balises rencontrées. Nous affichons, dans certains cas, la description associée à une balise, afin de rendre plus simple l'interprétation des figures.
Le cheminement afin de pratiquer une analyse est le suivant : nous regardons les données dans la liste des variants, puis nous extrayons celles concernées par l'analyse. Enfin, nous rangeons ces données extraites dans des dictionnaires, puis nous lançons une des fonctions d'affichage.

L'analyse de la qualité permet de visualiser par un histogramme, la moyenne de la qualité des variants pour chaque chromosome avec l'analyse « global », et la qualité de chaque variant détecté pour chaque chromosome avec l'analyse en choisissant un chromosome. Cette analyse permet d'avoir une vision globale ou spécifique sur la qualité des alignements des séquences des variants avec le génome de référence. Il est important pour un chercheur analysant ces données de savoir si ce sont de « vrais » variants, nous avons mis une barre horizontale du score conventionnel de qualité à 20 pour simplifier à l'utilisateur l'observation de variants de bonnes ou mauvaises qualité.

L'analyse « Filtre » permet de savoir le pourcentage de variants détectés qui ont passé ou non certains filtres spécifiques établies dans l'en-tête du fichier VCF. Cette analyse permet à l'utilisateur de savoir facilement quels filtres sont utilisés, et combien de variants ont été filtrés par ces filtres. Cela peut aider le chercheur à reconsidérer ses filtres ou à en utiliser d'autres selon ce qu'il veut faire de ces données.

L'analyse « Génome de référence » montre le pourcentage de chaque nucléotide (A,T,C,G) du génome de référence qui ont eu un variant. Cela permet de montrer si les variants génétiques relèvent de mutations ou variation structurale de nucléotide en particulier par rapport au génome de référence. Le type de nucléotide du génome de référence variant a une importance car selon les bases nucléotidiques il y a des différences de stabilité de l'ADN ou de l'ARNm qui peuvent s'effectuer. Ainsi que des modifications au niveau du méthylome pouvant avoir des effets régulateurs sur des régions génomiques par exemple.

L'analyse « Insertions et délétions » permet de connaître le nombre d'insertions ou de délétions de tous les variants, ou des variants d'un chromosome choisi. A savoir que cette analyse a des problèmes qui seront traitées dans la partie suivante du rapport, le problème principal étant que cette analyse comptabilise chaque nucléotide en plus ou en moins de la séquence du variant par rapport à la séquence de référence. Ainsi nous obtenons un nombre de nucléotides insérés ou enlevées par rapport au génome de référence, plutôt que les insertions ou délétions d'un ou plusieurs nucléotides détectées chez les variants. Nous voulions que cette analyse regroupe les différents types d'événements de variation structurale répertoriés chez les variants, y compris les substitutions nucléotidiques. Cela permettrait à l'utilisateur de connaître la proportion des types de mutations ou événements qui ont pu se produire pour obtenir les variants. Mais cette analyse prend en compte les différents variants relevant de tags qui sont répertoriés dans la partie « ##INFO » de l'en-tête de fichier VCF contenant des tags.

Problèmes techniques et difficultés :

La difficulté majeure au cours de projet a été de penser à la manière dont nous allions procéder pour stocker les données, de façons à pouvoir y accéder facilement. Nous avons opté pour des dictionnaires globaux. Ainsi, il est simple de les remplir et d'y accéder depuis n'importe quelle fonction dans le code. Un défi a été de réaliser l'interface graphique : à l'aide de la librairie tKinter, nous avons créé des "frames", que nous avons placées les unes à la suite des autres, toutes sur une frame principale (la fenêtre). Nous avons ensuite rajouté les objets (boutons, menus, textes) sur les frames adéquates. Nous avons opté pour l'option de la "facilité", et les objets ne sont pas ajoutés dynamiquement sur les frames. Ainsi, nous avons décidé que la taille de la fenêtre ne sera pas modifiable.

Il y eu également des difficultés liés à créer une analyse « général » de fichier VCF, car ceux-ci répertorient beaucoup de champs de méta-informations optionnelles, mais qui peuvent être importantes pour l'analyse des résultats.

Perfectionnements à effectuer :

Malgré le temps passé sur ce projet, nous avons encore des améliorations à faire, en particulier les points ci-dessous :

- Le code est fait de manière à ne pas créer de doublons lors de la lecture d'un fichier VCF, en particulier pendant la création des balises (les lignes commençant par "##"). Nous pensions que cela était nécessaire, dans le cas où l'entête pouvait être répété plus bas, comme dans les tests fournis. Après discussion, il est en fait impossible que celle-ci soit répétée, il aurait donc été plus habile au niveau des performances, et donc de la vitesse d'exécution, de ne pas vérifier les doublons à chaque ajout d'informations.

- Le bouton "Analyse Random" permet de lancer une analyse aléatoire, sur un chromosome aléatoire, ou bien tous les chromosomes. Un défaut de l'application est qu'il est possible, en cliquant sur le bouton, de tomber aléatoirement sur une analyse ne présentant pas de résultats, et donc d'ouvrir une fenêtre d'erreur. Cela peut être gênant, et il est possible de remédier à ce problème, mais cela impliquerait la modification d'une grande partie du code, nous avons donc décidé de laisser ce défaut en suspens.

Au niveau du code, et de son niveau esthétique, nous soulevons deux problèmes :

- Les fonctions pour pratiquer une analyse sur un chromosome en particulier, et une analyse sur tous les chromosomes sont très ressemblantes. Nous aurions pu optimiser ce côté-là du code, en créant une fonction unique travaillant sur un jeu de données, celui-ci étant créé par deux fonctions différentes. Cela nous posait problème pour les affichages graphiques cependant, donc nous avons abandonné cette idée. Nous aurions aussi pu ne pas proposer les fonctions d'analyse n'ayant pas de résultat, mais cela était encore plus fastidieux à faire.

- Certaines fonctions sont de taille trop longue, malgré nos efforts pour les scinder en plusieurs parties. En particulier l'analyse des insertions et délétions. Nous avons essayé cependant de coder des fonctions d'affichage globales, prenant pour arguments les textes à écrire.

Il faudrait faire que le code pour l'analyse des insertions et délétions comptabilisent toutes les insertions et délétions des variants, ainsi que les substitutions nucléotidiques et autres évènements répertoriés chez les variants. Comme dit ci-dessus à la fin de la partie des analyses réalisées, nous comptabilisons uniquement le nombre de nucléotides insérés ou supprimés chez les variants comparées au génome de référence. Une information qui biologiquement parlant n'a pas trop d'importance. Ainsi il faudrait faire de nombreux changements au code pour prendre en compte ces paramètres, ce qui aurait pris beaucoup trop de temps.

Une autre amélioration possible serait de pouvoir effectuer les analyses sur les variants qui relèvent d'une information spécifique marqué en en-tête du fichier VCF. Par exemple, si l'utilisateur souhaite analyser les variants qui relèvent d'allèles ancestraux.