Encadreur : Fanny Pouyet

Un projet réalisé par :

MABROUK Sermed

BOUNEGTA Mohamed

SAID Rachad

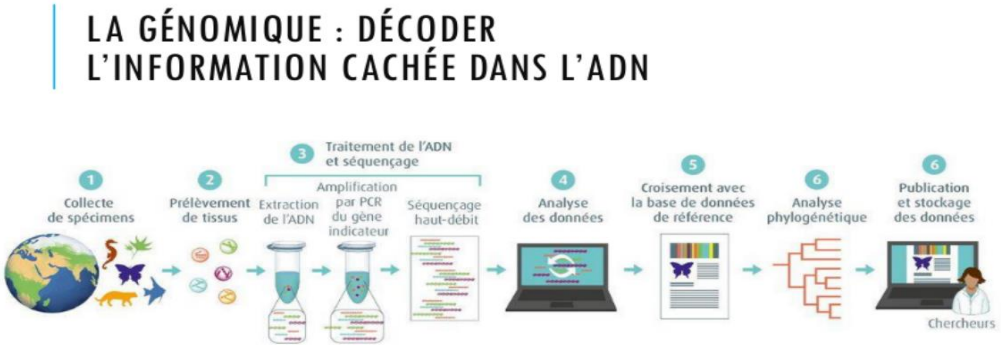
Projet Bio-Informatique

BUT :

Estimation de l'histoire évolutive de la levure Saccharomyces Cerevisiae

COMMENT ?

Création d'un Pipeline Bio-informatique de préparation et d'analyse de données de génomes.



Introduction :

Contexte :

Ce qui nous a ramené à ce qu'on est aujourd'hui est l'évolution, que ce soit historique, technologiques, ou même génétique. En effet, la diversité génétique est due à une évolution moléculaire qui a engendré des contraintes sur l'évolution des génomes et des espèces qu'on peut représenter et étudier aujourd'hui grâce aux arbres de coalescence et du séquençage de l'ADN.

Avec l'avènement du séquençage d'ADN à haut-débit, une masse phénoménale de données est disponible et son traitement nécessite de solides compétences informatiques

But :

Aujourd'hui on souhaite estimer les histoires évolutives de la levure naturelle et domestiquée S.CEREVISIAE, pour cela on aura besoin de résultats sous forme de Pipeline Bio-informatique de préparation et d'analyse de données de génomes.

Sujet :

Le but du projet est de créer un pipeline d'automatisation avec une implémentation de code adaptable.

Le principe d'automatisation :

Le pipeline est un mécanisme permettant d'accroître la vitesse d'exécution des instructions dans un micro-processeur. L'idée générale est d'appliquer le principe du travail à la chaîne à l'exécution des instructions. grâce au script qu'on a créer, les instructions sont exécutées les unes après les autres.

Afin de mettre en œuvre un pipeline, la première tâche est de découper l'exécution des instructions en plusieurs étapes à fin de rendre indépendantes les différentes étapes de traitement d'une instruction par le processeur ; Ici 4 grandes étapes avec des sous étapes.

Matériel :

Ce fut un projet ou le travail en équipe était une priorité, on était surtout dépendants les uns des autres, pour une bonne pratique, on a eu recours à Github, Parsec et Monday.com, Nous avons donc noté toutes les tâches à faire avec les deadlines et nos objectifs sur monday.com une plateforme pour gérer le projet avec le groupe d'une façon visuelle pour rapport d'avancement sous forme de cahier de texte, partagé nos ressources grâce à Github. Quand à Parsec, c'est l'application qui nous a permis d'éviter les problèmes de matériel rencontrés par un ou plusieurs membres du groupes .

Docker : Même si on a commencé à travailler sur une machine virtuelle qui est Conda, on a opté ensuite pour Docker qui nous a permis de travailler dans un conteneur comme un système d'exploitation mais plus léger avec plusieurs avantages : Très complet, beaucoup de dépôts maintenus, très stable. et l'un des ses plus grand inconvénient qui est la durée limitée de son image dans les temps ne pouvait pas nous poser de problème pour ce type de projet.

Github, Parsec, Monday, Github, les algos et les fichiers ressources de références dans la page du cours nous étaient suffisants comme matériel et méthodes pour bien réussir le projet et le compléter dans de bonnes conditions.

Difficultés rencontrées :

Cela n'a pas empêché les nombreux problèmes et difficultés par lesquelles on est passé dans lesquelles on trouve :

- Problème de gestion de mémoire occupé par les images qui parfois atteignaient des centaines de Giga-octets ce qui est énorme à cause du coût et de la performance nécessaire par l'ordinateur. Cela nous a coûté de refaire l'exécution des étapes à cause d'une saturation de disque dur (on estimer que 40 go était suffisante)
- Problème de Hardware. Java ou la fonction MarkDuplicate fait des Pass quand le processeur de l'utilisateur est utilisé à 100% et donc il continue son chemin sans retourner le résultat ce qui a provoqué à perdre 17 échantillons lors de cette étape on l'a ensuite réglé en ajoutant l'option XMG4 qui consister à le forcer, en temps normal un ordinateur ne consomme pas tous les cœurs du processeur mais L'hyper V gérer par WSL2 était configuré pour exécuter au plein potentiel de la machine
- - En utilisant un conteneurs, on avait de base un OS le plus léger qui permet le fonctionnement de la machine, ce qui veux dire qu'on a quasiment aucune librairie déjà installé ce qui fait que les erreurs levée par les algorithme qu'on avait utilisé (bwa,gatk) ne fournissais pas les requirements pour l'exécuter car ces librairies à leurs yeux sont déjà installer et y'aura pas beaucoup "d'issue" de la part des utilisateurs .
-

-Malgré que les algo utilisés était probablement les plus performant probablement les meilleurs leurs manipulations était encombrante pour certains

-bwa,bcftools,samtools :facile à mettre en place et à utiliser documentation et installation résumer dans un Readme bien détaillé on lui attribue la note de 10

-gatk : plus difficile à mettre en place nécessité une versions précise de java , leurs algorithme ou du moins ceux qu'on a utilisé ne retourner pas les causes de l'erreur (comme les fichiers.gz ne sont pas accepté quand on met un gz en input ect …) +le bug expliqué en dessus ,les packages sont mal organisé on s'est retrouver avec plusieurs MarkDuplicates sur le help(gatk) y compris celui de picard (ce n'est pas pour rien qu'il enregistre 1000 issue dans leurs git) , cependant la documentation est bien explicite, voir la meilleurs, avec des exemples on lui attribue la note de 7.

-A la rencontre d'un problème on revient au cours pour voir si on n'a pas sauté les étapes et si les étapes d'avant étaient correctes puis on s'est référé à la documentation du site et finalement sur les forums, l'avis des internautes bio-informaticiens nous ont beaucoup aider sur surmonter des bugs

Réalisation du projet :

1. La collecte des données

Pour cette étape on devrait s'assurer de ces points :

-Quel OS l'utilisateurs a (on a donc fait un script propre pour les gens qui utilise Windows, Mac et Linux)

-Si l'utilisateur avait assez d'espace dans son disque dur pour lui éviter de saturer son DD

-Si l'utilisateur avait déjà les fichiers (qu'il les a peut-être déjà téléchargé à la main)

-Une fois téléchargé, on s'est assuré de la bonne réception en comparant le md5 de l'host avec celui de l'ENA

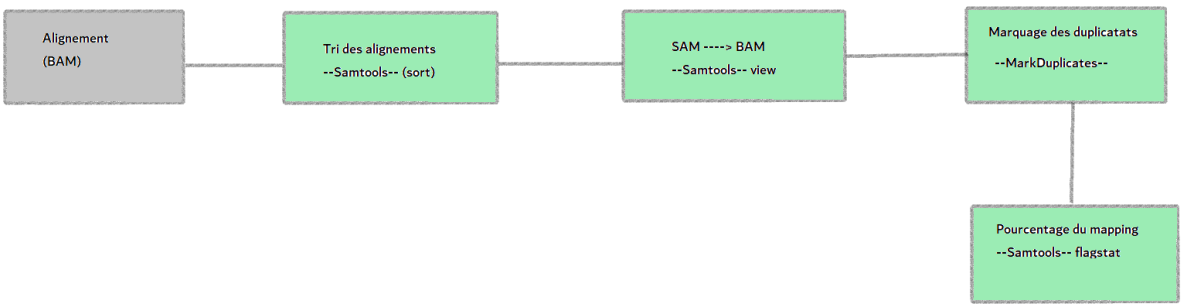
-Dézipper le bon génome référence via le script.

2. Le nettoyage : Mapping et Marquage

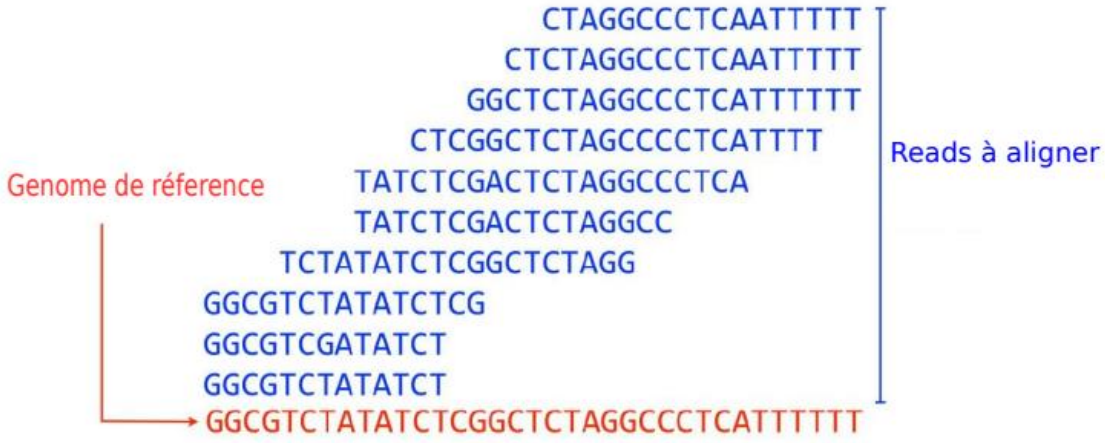
L'alignement :

➜ Les séquences obtenues doivent ensuite être remises dans le bon ordre par traitement bio-informatique, pour reconstituer le génome du patient. C'est l'étape d'alignement. Elle utilise comme modèle un génome humain de référence sur lequel le logiciel « aligne », de la manière la plus pertinente possible, tous les petits morceaux , en comparant une à une toutes les séquences d'ADN.

Pour cette étape on a eu recourt à l'algo BWA-MEM, qui est le plus récent, est généralement recommandé car il est plus rapide et plus précis. BWA-MEM a également de meilleures performances que BWA-backtrack pour les lectures Illumina de 70 à 100 bp et peut lire de longues séquences.



Mappage ; Alignement mais pas en ordre par rapport mappage par rapport au génome



Samtools sorted pour faire le tri des alignement.

SAMtools est un ensemble d'utilitaires pour interagir avec et post-traiter les alignements de lecture de séquences d'ADN courtes aux formats SAM, BAM et CRAM, écrits par Heng Li. Ces fichiers sont générés en sortie par des aligneurs de lecture courts comme BWA

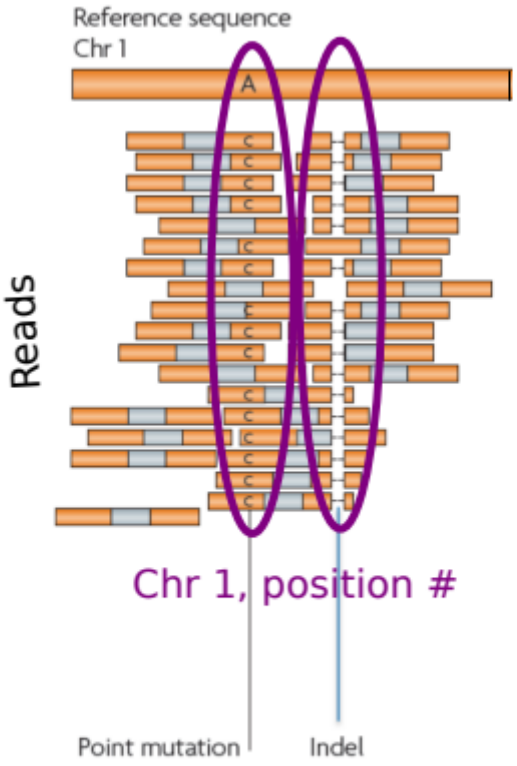
MarkDuplicates : marquer les Reads par rapport au chromosomes

Cet outil localise et balise les lectures en double dans un fichier BAM ou SAM, où les lectures en double sont définies comme provenant d'un seul fragment d'ADN

Détection des variants

Les algorithmes de calcul comparent, base par base, les séquences de l'ADN de la souche par rapport au génome de référence. Toutes les anomalies (appelées variants) sont repérées : modification d'une base, délétion ou insertion d'une séquence d'ADN, etc.

3. Identification des variants :



SNP/SNV **Indel**

Tout d'abord on fait un appel des SNP et des Indels (variants) grâce a la commande gatk HaplotypeCaller cette commande permet d'appeler simultanément des SNP et des indels via un assemblage local de novo d'haplotypes dans une région active .En d'autres termes, chaque fois que le programme rencontre une région montrant des signes de variation il rejette les informations existantes et réassemble complètement les reads dans cette région

Cela permet à HaplotypeCaller d'être plus précis lors de l'appel de région traditionnellement difficiles à appeler .

Comment il fonctionne ?

1 – Définir les regions actives

2- Déterminer les haplotypes par assemblage de la région active

3-Déterminer les probabilités des haplotyopes compte tenu des données lues

4-Attribuer des génotypes d'échantillons

GATK HaplotypeCaller :

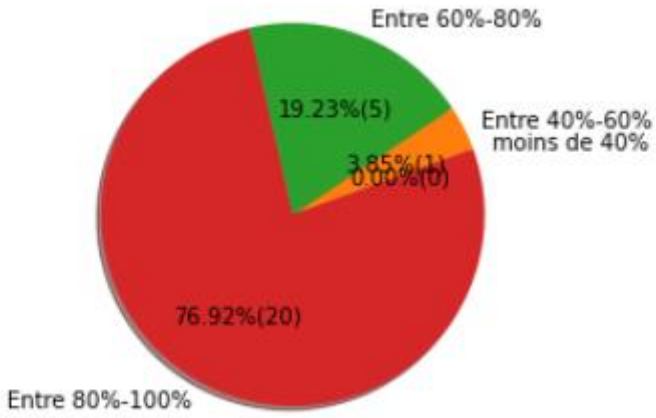
- GATK (Genome Analysis ToolKit) est une suite d'outils développée par le Broad Institute
- Bonne documentation (Best Practices)
- Permet la gestion d'analyse de plusieurs échantillons (format gVCF)

- Comporte une étape de réalignement local des indel.
- Algorithme bayésien

Ensuite nous avons créé une base de données décrivant les relations entités/attributs des échantillons et de chaque site grâce à la commande cependant nous avons décidé de créer une table VCF qu'avec les espèces bien mappés car nous avons favoriser l'approche conservatrice on veut être sûrs de ce qu'on a quitte à perdre de l'information pour cela on a procédé comme suit :

Nous avons fait une étude des espèces et nous les avons distribué selon leur pourcentage de mapping grâce a la commande samtools flagstat qui nous permet d'obtenir le pourcentage de mapping de chaque espèces on a obtenu ces resultats

```
{'moins de 40%': 0,
 'Entre 40%-60%': 1,
 'Entre 60%-80%': 5,
 'Entre 80%-100%': 20}
```



REPARTITION DES ESPECES SELON LEUR POUCENTAGE DE MAPPING

Par la suite nous avons créer un fichier 'concatened_map' dans lequel on a regroupe les espèces avec lesquels on voulait créer notre base de données puis on a limité l'etude des chromosomes jusqu'au chromosome ref|NC_001148| avant le chromosome mithocondrial nous avons reconnus ce dernier grâce a sa taille qui d'environ 86kb

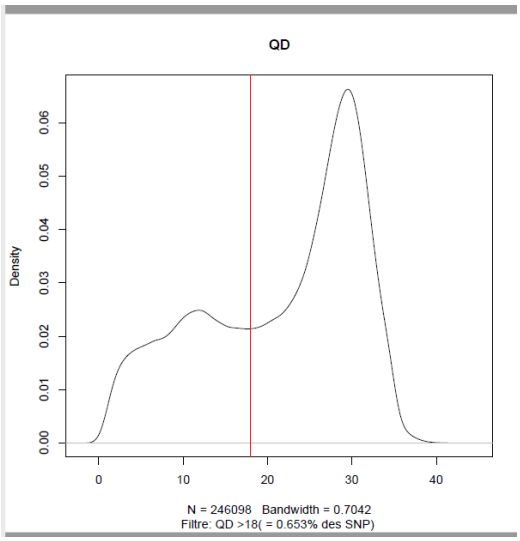
Et nous avons exécuter la commande gatk genomicsDBImport pour la création de la base de données

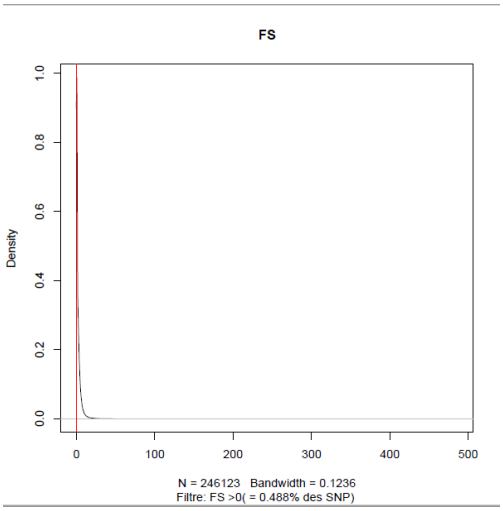
Puis gatk GenotypesGVF pour obtenir notre table VCF

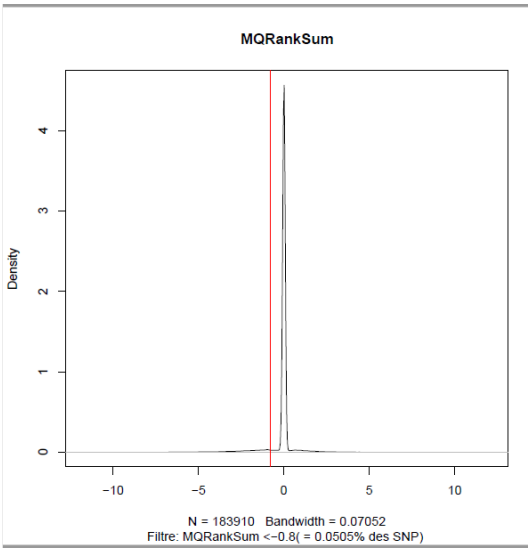
Par la suite nous avons sélectionner le variant qu'on voulait étudier (les SNP) en utilisant la commande Gatk SelectVariant et l'option '–select-type-to-include SNP ' afin de sélectionner que les

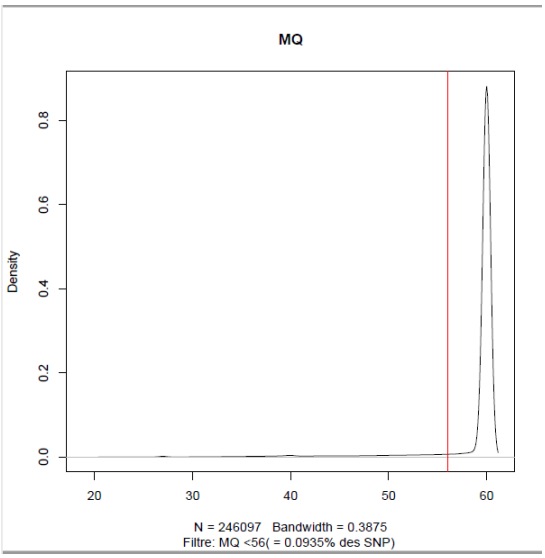
SNP

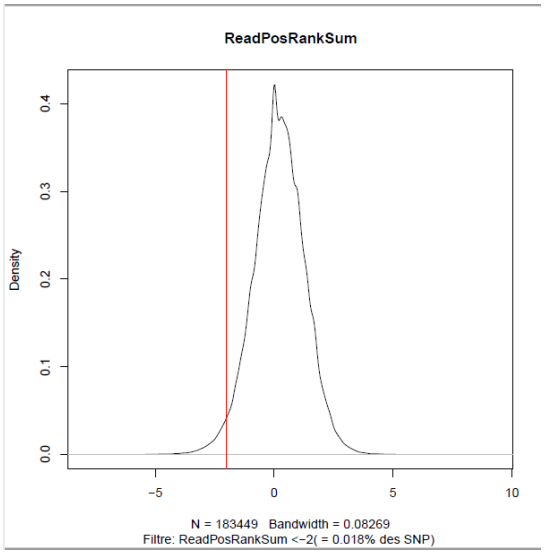
On a par la suite dessiner des graphes des 6 filtres qui détectent les SNP on a obtenu ces résultats

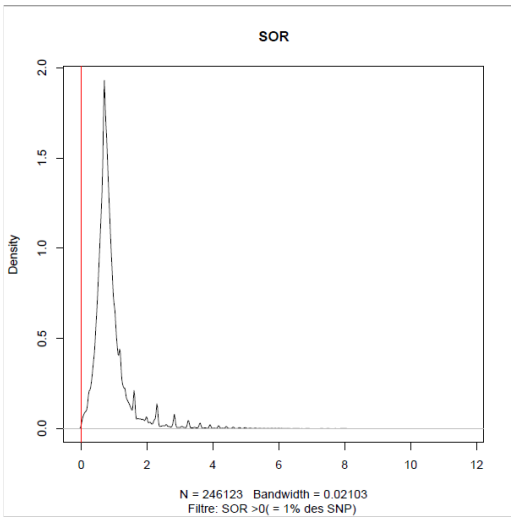

QD
Density
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0
10
20
30
40
N = 246098 Bandwidth = 0.7042
Filtre: QD >18(= 0.653% des SNP)


FS
Density
0.0
0.2
0.4
0.6
0.8
1.0
0
100
200
300
400
500
N = 246123 Bandwidth = 0.1236
Filtre: FS >0(= 0.488% des SNP)

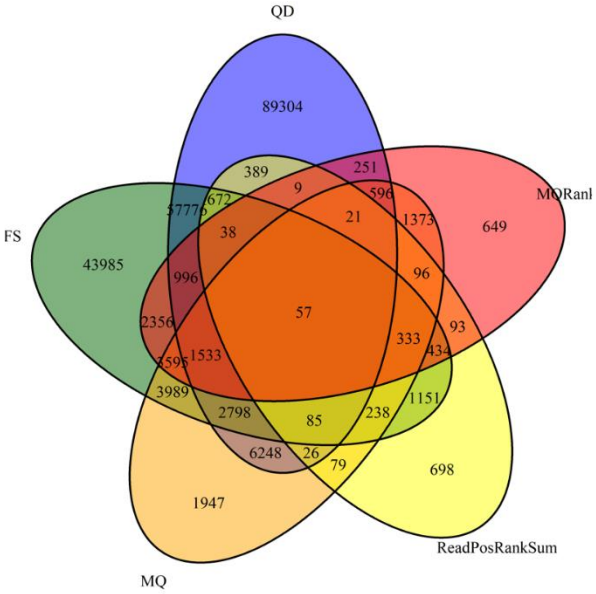

MQRankSum
Density
0
1
2
3
4
−10
−5
0
5
10
N = 183910 Bandwidth = 0.07052
Filtre: MQRankSum <−0.8(= 0.0505% des SNP)

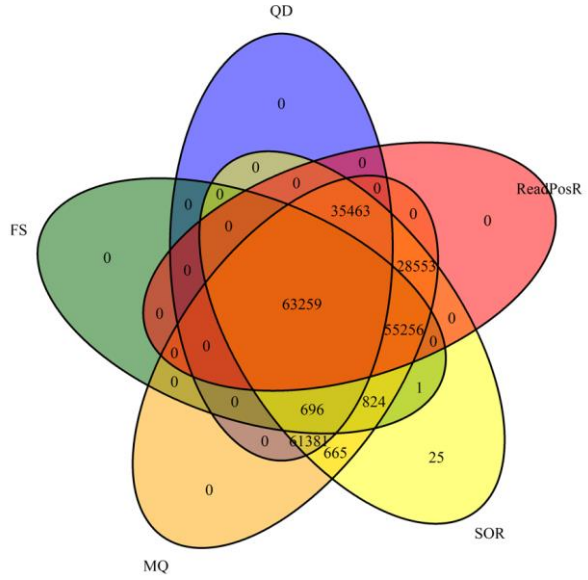






Et un diagramme de venn





4. **Analyses des résultats**

On constate que d'après le diagramme de venn le filtre qui nous donne le plus d'information c'est le filtre Quality Depth et que l'intersection des filtres et de 57 dans le premier diagramme ou il n'y a pas le filtre SOR .

Bilan :

Cette UE est pour nous l'une des plus intéressantes découverte en apprenant un domaine Bidisciplinaire, ou on a repris quelques connaissances du lycée en plus avancés qu'on a transposer dans notre spécialité qui est l'informatique ici de la science des données et des tests statistiques.

Conclusion :

On a était confronté à un travail de type de recherche/développement, c'était surtout des recherches de tutoriels informatiques et recherches de bibliographies, et bien évidemment vu que c'était un pipeline, le travail consistait plus à un avancement collectif plus qu'à une répartition des tâches vu que la nature du travail, ca a demandé une forte communication au sein du groupe.

Avis, motivation et ressentis :

Le travail sur ce projet s'est très bien passé ; cette première expérience en cette discipline était motivante et très intéressante, elle nous a vraiment plu et on sera encore plus déterminé pour une prochaine expérience.