TP MinION Roscoff

1. Connexion au cluster.
ssh <*login*>@bioinfo.sb-roscoff.fr
Réservation des ressources : qlogin
Activation de l'environnement nécessaire au TP : source $CONDA3/activate eba2017_long_read
Récupération des données pour le TP : cp /projet/sbr/ggb/minion/* .
Vérifier que la récupération s'est bien passée en utilisant la commande ls.
(6 fichiers devraient être listés)

Données MinION
----

2. Analyse de la qualité de séquençage MinION avec FastQC.
Lancement de FastQC :
fastqc 2d.fastq
Pour visualiser le résultat, il faut le récupérer sur votre ordinateur en local. Pour cela, ouvrez un terminal en local sur votre ordinateur et récupérez le fichier avec une commande de type :
scp <*login*>@bioinfo.sb-roscoff.fr:~/2d_fastqc.html .

Ouvrir ce nouveau fichier en local avec un navigateur et regarder la qualité des reads.

3. Assemblage des reads MinION avec Canu.
Lancement de Canu :
canu -assemble -p mito -d TP -genomeSize=69k -nanopore-corrected minion.fasta -useGrid=false

Les résultats de Canu seront écrits dans un nouveau répertoire qui s'appelle « TP ».

Vous pouvez déjà avoir quelques informations en suivant le log généré sur l'écran. Vous pouvez accéder à certaines caractéristiques des contigs construits simplement en allant récupérer les identifiants des contigs dans le fichier fasta généré par Canu grâce à la commande grep.

grep ">" TP/mito.contigs.fasta

Pouvez-vous expliquer que la taille du contig Canu (82 Kb) soit si différente de celle attendue (69 Kb) ?

4. Analyse de la séquence
Blast de la séquence contre elle-même sur le web.
https://blast.ncbi.nlm.nih.gov/Blast.cgi
Pour cela, récupérer le fichier mito.contigs.fasta en local sur votre ordinateur :
scp <*login*>@bioinfo.sb-roscoff.fr:~/TP/mito.contigs.fasta .

Ouvrir le navigateur sur le site du ncbi, cocher l'option « Align two or more sequences ». Puis pour la 1ere séquence, cliquer sur Parcourir et choisissez votre fichier mito.contigs.fasta. Faire la même chose pour la 2e séquence, puis lancer le megablast.
Une fois les calculs terminés, ouvrez l'onglet « Dot Matrix View » et concluez.

5. Trimming du contig d'après le résultat du blast
prinseq-lite.pl -trim_to_len 69000 -fasta TP/mito.contigs.fasta

6. Renommer le résultat du trimming
mv TP/mito_prinseq_good_*.fasta TP/mito_69k.fasta

Si vous avez le temps vous pouvez reproduire l'étape 4 avec ce fichier pour voir si le trimming s'est passé correctement.

7. Mapping bwa du résultat du polishing de Nanopolish sur l'assemblage de Canu
Fichier fourni : polished_nanopolish.fasta
Il est nécessaire d'indexer la référence avant de pouvoir faire un mapping :
bwa index TP/mito_69k.fasta
Lancer le mapping avec bwa par la commande :
bwa mem TP/mito_69k.fasta polished_nanopolish.fasta > outputNano_canu.sam

8. Transformation en _sorted.bam.bai pour IGV
Conversion du fichier au format sam en un fichier au format bam (fichier binaire) :
samtools view -b outputNano_canu.sam -o outputNano_canu.bam

Tri du résultat à partir du fichier binaire :
samtools sort outputNano_canu.bam -o outputNano_canu_sorted.bam

Indexation du fichier binaire trié (génération du .bai) :
samtools index outputNano_canu_sorted.bam

9. Visualisation IGV du mapping du résultat du polishing de Nanopolish sur l'assemblage Canu
Récupérer en local le fichier mito_69k.fasta (votre référence) ainsi que le résultat d'alignement outputNano_canu_sorted.bam et outputNano_canu_sorted.bam.bai
Lancer IGV.
Commencer par mettre le génome de référence : mito_69k.fasta
Cliquez sur Genomes → Load genome from file…
Chargez ensuite le bam en cliquant sur File → Load from file et choisir le fichier : outputNano_canu_sorted.bam
(attention le fichier bam et son index doivent se trouver au même endroit!)

=> observer le type de corrections faites par Nanopolish.

10. Mapping des reads MinION sur l'assemblage Canu pour comprendre la correction faite par Nanopolish
bwa mem -x ont2d TP/mito_69k.fasta minion.fasta > outputMinION_canu.sam

11. Visualisation IGV du mapping précédent
samtools view -b outputMinION_canu.sam -o outputMinION_canu.bam
samtools sort outputMinION_canu.bam -o outputMinION_canu_sorted.bam
samtools index outputMinION_canu_sorted.bam
Comme précédemment, récuperer les fichiers outputMinION_canu_sorted.bam et outputMinION_canu_sorted.bam.bai en local pour utiliser IGV et visualiser le résultat.

Observer le mapping des reads et les corrections faites par Nanopolish ?

Données Illumina
----

12. Evaluer la qualité des reads Illumina par FastQC
fastqc illu_R1.fastq
Récupérer le fichier html comme précédemment pour regarder le résultat.
=> observation : meilleure qualité que MinION. Pourquoi ne pas assembler en utilisant Illumina ?

13. Evaluation de l'assemblage des reads Illumina par Velvet
prinseq-lite.pl -fasta assemblage_illu.fasta -stats_all

=> observations : 38 séquences, la taille totale est bonne mais c'est beaucoup trop découpé. Même si l'assemblage est mauvais en terme de contiguité, pourquoi ne pas utiliser les reads Illumina pour corriger un assemblage MinION là où c'est possible ?

14. Correction de l'assemblage à l'aide des données Illumina
Lancement du logiciel Pilon :

- nécessité d'un bam (résultat de mapping des reads Illumina contre l'assemblage issu de Canu) trié et indexé → lancement de bwa

i) indexation de la référence (déjà fait à l'étape 7)

ii) mapping des R1 et R2 Illumina

```
bwa aln TP/mito_69k.fasta illu_R1.fastq > illu_R1.sai
bwa aln TP/mito_69k.fasta illu_R2.fastq > illu_R2.sai
bwa sampe TP/mito_69k.fasta illu_R1.sai illu_R2.sai illu_R1.fastq illu_R2.fastq > illu_canu.sam
```

iii) sam to bam_sorted.bai

```
samtools view -b illu_canu.sam -o  illu_canu.bam
samtools sort illu_canu.bam -o illu_canu_sorted.bam
samtools index illu_canu_sorted.bam

pilon --genome mito_69k.fasta --frags illu_canu_sorted.bam
```

15. Mapping du résultat de Pilon sur Canu
bwa mem TP/mito_69k.fasta pilon.fasta > outputPilon_canu.sam

16. Transformation en *_sorted.bam.bai

```
samtools view -b outputPilon_canu.sam -o  outputPilon_canu.bam
samtools sort outputPilon_canu.bam -o outputPilon_canu_sorted.bam
samtools index outputPilon_canu_sorted.bam
```

17. Visualisation sous IGV des différents mapping
Récupérer les fichiers illu_canu_sorted.bam et outputPilon_canu_sorted.bam pour visualiser sous IGV.
Chargez les bam : illu_canu_sorted.bam, outputPilon_canu_sorted.bam
en cliquant sur File → Load from file…

=> observations : relier les différences entre les reads illumina et la correction faite par Pilon sur Canu.

Références des outils utilisés

Contrôle de la qualité des séquences :
FastQC : https://www.bioinformatics.babraham.ac.uk/projects/fastqc/
Assemblage de reads MinION :
Canu : http://canu.readthedocs.io/en/latest/tutorial.html#canu-the-command
Boîte à outils pour gérer des séquences :
prinseq-lite.pl : http://prinseq.sourceforge.net/manual.html
Mapper :
bwa : http://bio-bwa.sourceforge.net/bwa.shtml
Boîte à outils pour gérer des résultats de mapping :
samtools : http://www.htslib.org/doc/samtools.html
Amélioration d'un assemblage de reads MinION grâce à des reads Illumina :

pilon : https://github.com/broadinstitute/pilon/wiki
Visualisateur :
IGV : http://software.broadinstitute.org/software/igv