TP/Lab session: RNA-seq Pipelines

Daniel Gautheret - I2BC

5/9/2022

TP/Lab session: RNA-seq Pipelines

Author: Daniel Gautheret - I2BC

Human Differentially Expressed Genes during the EMT Process

1/ The EMT dataset

The RNA-seq data comes from a study of a cell transformation process called epthithelium-mesenchymal transition (EMT) by Yang et al. (2016). A transformation process called EMT was induced by ectopic expression of Zeb1 in a non small cell lung cancer cell line (H358). The authors obtained RNA-seq data over a time course of 7 days starting from uninduced cells and every day up to 7 days of induction.

OBJECTIVE: Our goal is to observe genes that are either up or down regulated during the process, that is between Day 0 (uninduced cells) and Day 7 (7 days after induction).

Sequence libraries are polyA+, pair-end 2x100nt, each in biological triplicate. Sequencing is performed on a Illumina HiSeq 2500.

Paper: Yang Y, Park JW, Bebee TW, Warzecha CC, Guo Y, Shang X, Xing Y, Carstens RP. Determination of a Comprehensive Alternative Splicing Regulatory Network and Combinatorial Regulation by Key Factors during the Epithelial-to-Mesenchymal Transition. Mol Cell Biol. 2016. 36:1704-19 (http://www.ncbi.nlm.nih.gov/pubmed/?term=27044866)

Fastq files were obtained here: http://www.ncbi.nlm.nih.gov/sra?term=SRP066794 (about 72Mx2 reads per file)

For this lab session, we sampled about 0.5% of the total RNA-seq reads. All retained reads are from a single chromosome (chr18). We mapped all reads against the HG19 human genome using the STAR program, filtered reads mapping to chromosome 18 using the grep command, and filtered out 50% of the remaining reads using Samtools. This represents 0.5% of total reads, thus actual runtimes and space requirement would be up to 200 times higher than in our exercices.

2/ Récupération des fastq

Connectez-vous à votre VM. Creez un dossier de travail pour y placer tous les fichiers de ce projet. Placez-vous dans ce dossier.

Récupérez les fichiers fastq R1 et R2 dans: http://rssf.i2bc.paris-saclay.fr/X-fer/AtelierNGS/TPrnaseq.tar.gz (Utilisez wget directement depuis la VM IFB) Notez les tailles des fichiers et nombres de reads moyens approximatifs pour chacune des deux conditions.

```
ssh [user@yourVM-IP-address]
mkdir TPRNAseq
cd TPRNAseq
wget  http://rssf.i2bc.paris-saclay.fr/X-fer/AtelierNGS/TPrnaseq.tar.gz
tar -zxvf TPrnaseq.tar.gz
```

Pour la suite du TP il est recommandé de réaliser d'abord l'ensemble de la procédure dans un terminal sur un échantillon apparié (reads R1 et R2). Une fois que toute la procédure fonctionne pour un échantillon, l'automatiser pour tous les échantillons avec un script shell.

3/ Contrôle qualité

Lancez la commande fastqc pour un fichier R1 et un fichier R2. Visualisez les différents graphiques. Notez-vous une différence de qualité entre R1 et R2?

A l'aide d'une seule ligne de commande, traitez par fastqc l'ensemble des fichiers fastq. (fastqc est capable de prendre une liste de fichiers avec wildcard "*")

4/ Trimming et retrait des adaptateurs

Utilisez trimmomatic pour éliminer les séquences de basse qualité en extrémité des reads. Nous n'avons pas besoin de retirer les adaptateurs avec les banques EMT.

```
trimmomatic PE name.R1.fastq name.R2.fastq -baseout name.fastq  LEADING:20 TRAILING:20 MINLEN:50
```

Avec ces paramètres, Trimmomatic retire les bases des extrémités 5' et 3' tant que leur qualité est inférieure à 20. Si le read final a une taille inférieure à 50 il est éliminé. 4 fichiers sont produits. 2 fichiers R1,R2 avec les reads nettoyés, et 2 fichiers R1,R2 avec les reads éliminés. l'argument -baseout donne le prefixe.suffixe des fichiers produits.

Relancez fastqc pour vérifier l'amélioration de la qualité et le nombre de reads perdus par la procédure.

5/ Mapping

5.1 Recupération du Génome et Annotation

Les téléchargements peuvent être faits en ligne de commande avec wget ou curl.

Téléchargez le fichier génome humain (chr18) au format fasta sur le site UCSC:

```
http://hgdownload.soe.ucsc.edu/goldenPath/hg19/chromosomes/chr18.fa.gz
```

Décompressez le fichier fasta.

Téléchargez l'annotation Gencode au format gtf sur le génome version Hg19/GRCh37 ici:

```
ftp://ftp.ebi.ac.uk/pub/databases/gencode/Gencode_human/release_24/GRCh37_mapping/gencode.v24lift37.bas
```

Décompressez le fichier gtf.

5.2 Indexation du génome avec STAR

Attention: cette étape nécessite 16Go de RAM pour indexer un chromosome (32Go pour le génome entier). Si vous n'avez pas cet espace, récupérez l'index précalculé via l'enseignant ou un autre étudiant.

Réalisez l'indexation avec les paramètres suivants de STAR:

```
STAR --runMode genomeGenerate --runThreadN [nbre de coeurs] \
  --genomeDir [nom du dossier index] \
  --genomeFastaFiles [genome fasta] \
  --sjdbGTFfile [annotation gtf]
```

5.3 Mapping avec STAR (one-pass)

Créez un dossier pour les sorties de STAR, puis lancez STAR avec les paramètres:

```
STAR --runThreadN [nb core] --outFilterMultimapNmax 1\
  --genomeDir [votre dir pour les index genome] \
  --outSAMattributes All --outSAMtype BAM SortedByCoordinate \
  --outFileNamePrefix [/votre/dossier/resultat/prefixedubam] \
  --readFilesIn [fastq R1] [fastq R2]
```

Vérification de la bonne exécution du mapping

- le fichier BAM est-il bien produit et non vide?

Vous trouverez dans le fichier log final (*Log.final.out) les informations suivantes:

- Nb total de reads
- Nb de reads alignés de facon unique
- Nb de sites d'épissage canoniques (GT/AG)

Retrouvez-vous bien le nombre de reads du fastq initial?

6/ Tri et indexation du fichier BAM

Pour être utilisés par IGV et FeatureCounts, les BAM doivent être triés et indexés. Le tri est déjà fait par STAR (option SortedByCoordinate). On utilisera "samtools index" pour produire l'index.

```
samtools index [/path/to/bamfile]
```

(Option: allez à section "Visualisation IGV)

7/ Boucler sur tous les échantillons

Dans un script shell, faites une boucle pour répéter l'ensemble de la procédure sur les 6 échantillons (à partir des résultats de trimmomatic) Pour boucler sur les echantillons, on peut stocker les 6 préfixes des fastq dans une liste:

```
sample_list=(pref1 pref2 pref3...)
for i in ${sample_list[*]};do
  programme [bla bla ...] ${i}.fastq [... bla bla]
done
```

ou bien dans un fichier:

```
for i in `cat samples.txt`
do ...
```

8/ Comptage

Pour chaque banque RNA-seq nous voulons compter les reads alignés sur chaque gene.

Featurecount s'installe avec le package 'subreads' (pas de conda trouvé, utilisez apt-get install subread)

Parametre de featureCounts pour banques paired-end multiples et comptage au niveau du gene:

```
featureCounts -p -t exon -g gene_id -a [annotation.gtf] \
   -o [output.txt] [library1.bam] [library2.bam] [library3.bam]...
```

(featurecounts est capable de prendre une liste de fichiers avec wildcard "*")

Avec la commande Perl suivante, fabriquez une table d'équivalence entre noms ENCODE et noms HUGO, à partir du fichier gtf:

```
perl -ne 'print "$1 $2\n" if /gene_id \"(.*?)\".*gene_name \"(.*?)\"/' \
   [gtf file] | sort | uniq
```

Après avoir trié le fichier produit par featurecounts et cette liste d'equivalence, joignez ces deux fichiers sur la premiere colonne avec la commande join. Filtrez sur le chr18.

(dans le script ci-dessous on admet que le résultat de l'étape précédente a été sauvegardé dans encode-to-hugo.tab)

```
sort counts.txt > temp1
sort encode-to-hugo.tab > temp2
join temp1 temp2 |grep "chr18" > temp3
```

A l'aide de awk, créez une table de comptage du type:

```
RALBP1 4392 8049 7116 6690 3623 3574 3800
RAB27B 7281 1581 1356 1378 334 347 375
DSG2 6190 8051 6929 6741 2468 2450 2795
NEDD4L 10066 1746 1557 1571 1104 1150 1212
LAMA3 11037 6229 5526 5152 2083 2108 2188
SERPINB3 2182 0 0 1 0 0 0
```

Rappel: structure de la commande awk pour imprimer les colonnes 5 et 1:

```
awk '{print $5 " " $1}' [inputfile] > [outputfile]
```

C'est ce type de table qui sera utilisé pour identifier les gènes differentiellement exprimés dans la suite du TP, sous R.

9/ Script final

Intégrez l'ensemble du pipeline dans un script shell final. N'incluez pas les installations et téléchargements dans le script. On peut supprimer aussi l'étape fastqc et indexation du génome. Pour un script plus élégant, vérifiez l'existence des dossiers dans le script et à créez-les si nécessaire:

```
if [ ! -d index ];then
  mkdir index
fi
```

Votre pipeline entier doit pouvoir fonctionner en une seule ligne de commande. Vérifiez.

10/ Visualisation d'alignement sous IGV

(Nécessite l'installation de IGV sur station locale et un accès Internet car IGV télécharge automatiquement les génomes de référence)

Pour installer IGV sur station PUIO:

- Site web Broad IGV: binary install
- decompresser l'archive: unzip IGVxxxx.zip
- executer igv.sh: [chemin]/igv.sh

Si message d'erreur sur genome hg19 ne pouvant etre chargé:

- copier le genome hg19 depuis /partage/public/gautheret/NGS/IGV dans votre dossier ~/igv/genomes
- Puis depuis IGV, choisissez ce fichier dans load genome file

Visualisation des données RNA-seq sous IGV:

- Récupérez sur votre station locale le fichier .bam et .bai d'un échantillon "Day0" et un échantillon "Day7".
- Lancer IGV
- Vérifiez que le genome (HG19) et l'annotation (refseq) soient utilises par defaut. Sinon installez-les depuis IGV.
- Chargez un fichier BAM indexé «Day0»
- Naviguez sur le chromosome 18
- visualisez l'orientation des reads (color by "first of pair")
- Visualisez: Introns, exons, annotation étendue (expanded)

Loci intéressants: - chr18:19449740-19449780 (délétion) - chr18:32827183 (SNP) - Cas d'épissage alternatif dans les gènes ZNF397, C18orf21, SLC39A6 - Un gène non annoté vers 33762000

Visualisez le gene CDH2 sur chr18. C'est un indicateur de l'EMT. Il doit etre surexprimé par un facteur ~2 dans les conditions "Day7". (pour voir cela chargez un BAM "Day7")