TP/Lab session: Gene expression counts from RNA-seq data

Daniel Gautheret - I2BC & Bioinformatics Core, Gustave Roussy

17/2/2019

1/ The EMT dataset

The dataset comes from 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)

Sequence libraries are polyA+, stranded, paired-end 2x100nt, sequenced on a Illumina HiSeq 2500.

We retrieved Fastq files for conditions: Day_0 (uninduced) and Day_7 (7 days after induction), each in triplicate. 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. The resulting BAM files contain about 0.5% of the total RNA-seq reads.

OBJECTIVE: Our goal is to measure gene expression in each of the 6 BAM files (3x Day_0 and 3x Day_7).

2/ Retrieving BAMs

Connect to your VM. Create a work directory for all files in this project. cd to this directory, then retrieve all R1 and R2 files from:

https://transfert.u-psud.fr/s138my8c/download

(use wget)

```
ssh [user@yourVM-IP-address]
mkdir TPRNAseq
cd TPRNAseq
wget  https://transfert.u-psud.fr/s138my8c/download -O emt.tar.gz
tar -zxvf emt.tar.gz
```

Using ‘samtools stats’, check the number of mapped reads for some of your BAM files…

```
samtools stats Day_7_1_chr18.sampled.bam | less
```

3/ Visualizing RNA-seq data with IGV:

- Retrieve on your local workstation .bam and .bai files for a “Day0” sample and “Day7” sample.
- Launch IGV
- The genome version and annotation should be set to HG19 and refseq. If these are not set properly, install them using the IGV menus.
- Load an indexed BAM file for «Day0»
- Move to chromosome 18 and zoom on a gene.
- Visualize read orientation (color by “first of pair”)
- Visualize: introns, exons, expanded Refseq annotation

Interesting loci:

- chr18:19449740-19449780 (deletion)
- chr18:32827183 (SNP)
- Alternative splicing in genes ZNF397, C18orf21, SLC39A6
- An unnanotated gene around 33762000

Check gene CDH2. It is an EMT hallmark. It should be overexpressed by a factor ~2 in condition “Day7”. (you need to load a BAM for “Day7”)

4/ Counting

4.1/ Retrieval of Gencode annotation

Using wget, download Gencode gtf annotation version Hg19/GRCh37 here:

```
ftp://ftp.ebi.ac.uk/pub/databases/gencode/Gencode_human/release_24/GRCh37_mapping/gencode.v2
```

4.2/ FeatureCount

For each RNA-seq library, we want to count reads aligned on each gene with FeatureCount.

Featurecount is part of the Subreads package (not available in conda, use instead 'sudo apt-get install subread' if not installed)

Parameterization of featureCounts for paired-end libraries and gene-level count (output is written to [counts.txt]):

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

(featurecounts can also take a file list using wildcard symbols "*")

4.3/ Building a count table with gene symbols

Gene names in the gtf file are in ENSEMBL format (ENSG00000017797.11). No biologist likes that. Fortunately, a more practical gene name (HUGO gene symbol) is found on each line under "gene_name". Use the following Perl command to build an equivalence table between ENSEMBL and HUGO names:

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

Sort the featureCount table and the equivalence table, and join them (by their 1st column). Filter (grep) the result to keep only chr18 genes. Finally, use awk to extract gene names and counts in this form:

```
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
```

```
sort counts.txt > temp1
sort ensembl-to-hugo.tab > temp2
join temp1 temp2 |grep "chr18" > temp3
awk '{print $13 " " $7 " " $8 " " $9 " " $10 " " $11 " " $12}' temp3 > count.table
```

This type of table can then be loaded into R for differential gene expression analysis, for instance with DESeq or EdgeR.