RNASeq Analysis - Differential gene expression analysis at gene level

Aminata NDIAYE

15 octobre 2017

The goal of this analysis is to perform a differential expression analysis at gene level

1. Download 4 sequence datasets deposited to the EBI ENA:

Sequence datasets were taken into run section by typing in the search bar (http://www.ebi.ac.uk/ena/data/view) sequence IDS.

2. Extract fastq files

Fastq files have been extracted through their ftp addresses found in TXT results. After connecting with putty on a remote server, I used the following commands to download the fastq files into a folder.

ERR990557s.fastq

- *wget ftp.sra.ebi.ac.uk/vol1/fastq/ERR990/ERR990557/ERR990557.fastq.gz*

ERR990558s.fastq

- *wget ftp.sra.ebi.ac.uk/vol1/fastq/ERR990/ERR990558/ERR990558.fastq.gz*

ERR990559s.fastq

- *wget ftp.sra.ebi.ac.uk/vol1/fastq/ERR990/ERR990559/ERR990559.fastq.gz*

ERR990560s.fastq

- *wget* ftp.sra.ebi.ac.uk/vol1/fastq/ERR990/ERR990560/ERR990560.fastq.gz

3. For each file, select 8,000,000 (8 millions) of sequence reads and generate the following sample files:

To select only 8 millions of sequence reads I firstly use gzip to dezipp the original fastq files. Then I used the seqtk tools to sample the amount of sequence reads wanted in my fastq file. These are the following commands used (examples for ERR990557)

- Dezipping zipped files:

gzip -d ERR990557.fastq.gz

- Sampling 8 millions of sequence reads:

  sqtk sample ERR990557.fastq 8000000 > ERR990557.selected.fastq

4. Align these read datasets to the reference genome by any appropriate mean, and generate a sorted bam alignment file.

I first downloaded the Drosophila reference genome through the flybase database. Subsequently I used bwa mem for the alignment of reads. For their ordination I used the tool samtoolssort (ordination by coordinates) which also allowed me to convert the sam files into bam.

5. Count reads aligning to genome's genes by any appropriate mean

To count reads aligned to each genome gene, I used the samtools tool. Through the following commands, the numbers of the mapped reads found are as follows.

- *samtools view -F 0x904 -c ERR990557.bam*

  3349122

- *samtools view -F 0x904 -c ERR990558.bam*

  3639848

- *samtools view -F 0x904 -c ERR990559.bam*

  2999175

- *samtools view -F 0x904 -c ERR990560.bam*

  3465490

6. Perform a statistical differential expression analysis and report using any appropriate figures(s)/graph(s)

To perform a statistical differential expression analysis, I used the DESeq packages. Firstable I generated a count table from the datasets with kallisto tool throught this following command line:

- *kallisto quant -i dmel-all-CDS-r6.17.fasta.fai -o output --single ERR990557.selected.fastq ERR990558.selected.fastq ERR990559.selected.fastq ERR990560.selected.fastq*

The count table generated were then loaded into Rstudio.

The following part summarizes my R script, used to perfom differential expression analysis at gene level.

```r
#Installing DESeq and Biobase ; Loading libraries:
source("https://bioconductor.org/biocLite.R")
biocLite("DESeq2")
library(DESeq2)

#source("https://bioconductor.org/biocLite.R")
#biocLite("Biobase")
library (Biobase)

#Loading data files
datapath <- file.path("C:\\Users\\Aminata\\Desktop\\ARTbio")
count_table <- file.path(datapath, "count_table.tsv")
CountTable <- read.table(count_table, header = TRUE, row.names = 1)

#Clinical conditions setting and sample choice
CountTableDesign <- data.frame(row.names = colnames(CountTable), condition =
c("untreated","untreated", "treated", "treated"), libType = c("single-end",
"single-end", "single-end", "single-end"))
CountTableDesign
```

```
##           condition    libType
## ERR990557 untreated single-end
## ERR990558 untreated single-end
## ERR990559   treated single-end
## ERR990560   treated single-end
```

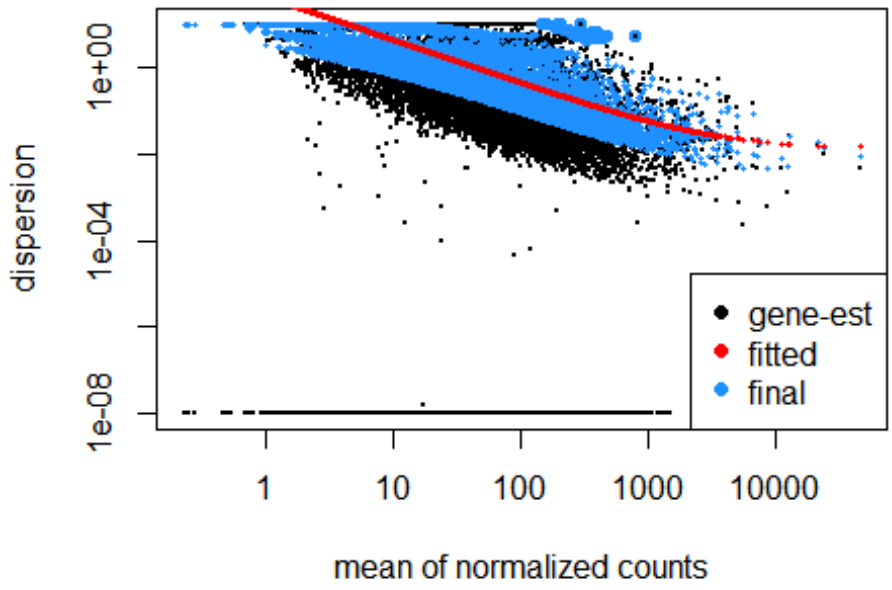
```r
pairedSamples <- CountTableDesign$libType == "single-end"
countTable <- CountTable[, pairedSamples]
condition <- CountTableDesign$condition[pairedSamples]

condition <- factor (c("untreated", "untreated", "treated", "treated"))
coldata <- data.frame(row.names = colnames(countTable), condition)
cds <- DESeqDataSetFromMatrix(countData=countTable,colData=coldata,
design=~condition)
cds <- estimateSizeFactors(cds)
sizeFactors(cds)
```

```
## ERR990557 ERR990558 ERR990559 ERR990560
## 1.0126253 1.0442851 0.9025858 1.1092029
```

```r
#Overview of the normalized data by dividing each column of the count table
by the size factor of this column to bring the count values to a common
scale.
head (counts (cds, normalized=TRUE))
```

```
##            ERR990557 ERR990558 ERR990559  ERR990560
## Nep3-PA     5.925193  10.53352   67.5836 48.6836102
## Nep3-PB     5.925193  10.53352   67.5836 48.6836102
## Nep3-PC     5.925193  10.53352   67.5836 48.6836102
## CG9570-PA   0.000000   0.00000    0.0000  0.0000000
```

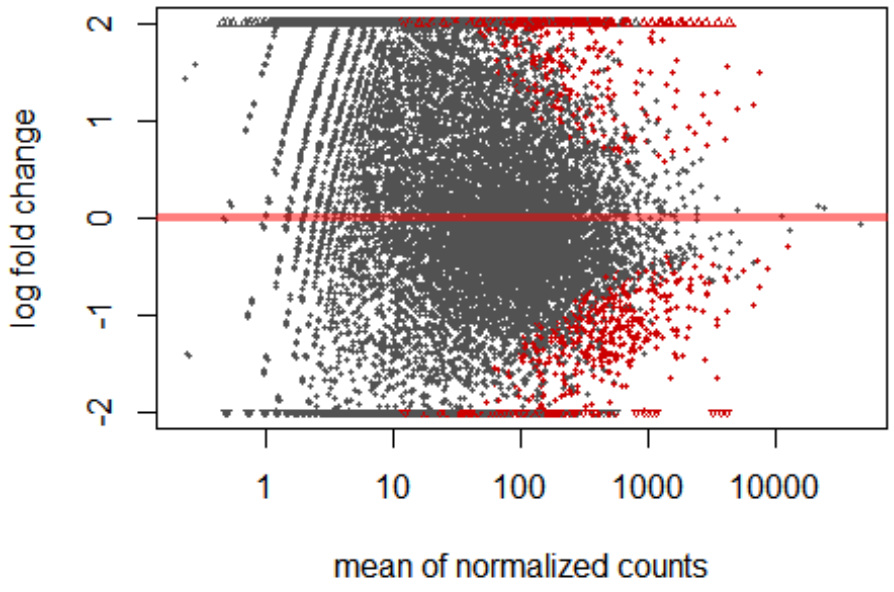
```
## Or19b-PA    0.000000   0.00000    0.0000  0.9015483
## CG15322-PB  0.000000   0.00000    0.0000  0.0000000

#Variance estimation
cds <- estimateDispersions(cds)

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

#Overview of the dispersion estimation
plotDispEsts(cds)
```




```
#Perform a differential analysis gene expression
dds <- DESeq(cds)

res <- results(dds, contrast = c("condition", "treated", "untreated"))

#View first lines of DESeq results
head(res)

## log2 fold change (MLE): condition treated vs untreated
## Wald test p-value: condition treated vs untreated
## DataFrame with 6 rows and 6 columns
##              baseMean log2FoldChange     lfcSE      stat     pvalue
##             <numeric>      <numeric> <numeric> <numeric>  <numeric>
```

```
## Nep3-PA    33.1814813       2.817855 0.9395395 2.9991871 0.00270701
## Nep3-PB    33.1814813       2.817855 0.9395395 2.9991871 0.00270701
## Nep3-PC    33.1814813       2.817855 0.9395395 2.9991871 0.00270701
## CG9570-PA   0.0000000             NA        NA        NA          NA
## Or19b-PA    0.2253871       1.440108 4.9900183 0.2885977 0.77288924
## CG15322-PB  0.0000000             NA        NA        NA          NA
##                   padj
##              <numeric>
## Nep3-PA    0.03619879
## Nep3-PB    0.03619879
## Nep3-PC    0.03619879
## CG9570-PA          NA
## Or19b-PA           NA
## CG15322-PB         NA
```

```r
#Ouput the results as a matrix table
write.table(res,
file="C:\\Users\\Aminata\\Desktop\\ARTbio\\RNAseq_analysis.txt")

#MA-plot
plotMA(res, ylim=c(-2,2))
```




```r
#Filter for significant genes, accrodinf to some chosen p-adj
res_significance <- subset(res, padj < 0.01)

#Output gene likely to be differentially expresses with padj > 0.01
```

```
write.table(res_significance,
file="C:\\Users\\Aminata\\Desktop\\ARTbio\\Differentially_expressed.txt")
```