RNA-seq hands-on tutorial using Chipster: ENCODE dataset

Eija Korpelainen, CSC – IT Center for Science, chipster@csc.fi

In this tutorial you start with raw reads (in fastq files), and learn how to check the read quality, trim bad quality bases, check the strandedness of the data, align reads to genome, and count reads per genes. Then you combine count files for all the samples in one table, and describe your experimental setup using the phenodata file. You also learn how to check coverage uniformity, and whether novel splice junctions were found. Finally, you detect differentially expressed genes and learn how to visualize reads in the genomic context using a genome browser.

The data is a small subset of single end RNA-seq reads from two human cell lines, h1-hESC and GM12878. Note that when analyzing differential expression you should always have at least 3 biological replicates! We use this small dataset for the first steps of the analysis for the interest of time, but we'll move to another dataset with replicates for the actual differential expression analysis later.

1. Start Chipster and open a session

Go to **http://chipster.csc.fi/**, and **Launch Chipster**. Log in with your username and password.
Select **Open example session** and **course_RNAseq_ENCODE**. This session has two fastq files. Note that normally fastq files are zipped and Chipster can use them like that.

2. Quality control of reads with FastQC

Select **hESC.fastq**, the tool **Quality control / Read quality with FastQC** and click **Run** (please do not analyze GM12878.fastq yet, we will use it later on for a workflow). Select the **resulting html file** and viewing option **Open in external web browser**.
-How many reads are there and how long are they?
-Is the base quality good all along the reads?

3. Trim reads based on base quality with Trimmomatic (note that this step is optional in real life)

Select again **hESC.fastq** and the tool **Preprocessing / Trim reads with Trimmomatic** and set the parameters: **Minimum quality to keep a trailing base** = **5** and **Minimum length of reads to keep = 50**.
-How many reads get discarded (select the file **hESC-trimlog.txt** and **View text**)?

Select **hESC_trimmed.fq.gz** containing trimmed reads and run the tool **Read quality with FastQC** as before.
-Does the base quality look better now?

4. Check the strandedness of the reads

Select **hESC_trimmed.fq.gz** and run the tool **Quality control / RNA-seq strandedness inference and inner distance estimation using RseQC** (check that **genome = Homo_sapiens.GRCh38.95**). Open the resulting **experiment_data.txt**.
-Is the data stranded? From which strand are the reads from? Mark down the parameters for HISAT2 and HTSeq.

5. Align reads to reference genome using HISAT2

Select **hESC_trimmed.fq.gz** and run the tool **Alignment / HISAT2 for single end reads** setting **genome = Homo_sapiens.GRCh38.95** and **Library type = fr-secondstrand**.
-What was the overall alignment rate and how many reads have multiple alignments (hisat.log)?

6. Perform alignment level quality check with RseQC

Select **hESC.bam** and the tool **Quality control / RNA-seq quality metrics with RseQC**. In parameters set **organism = Homo_sapiens.GRCh38.95**.

-Inspect the result file **hESC_rseqc.txt**. How many alignments does the BAM file contain? Is the tag (~read) density higher in exons than in introns?
-Inspect the result file **hESC_rseqc.pdf**. Is the coverage uniform along transcripts (check the first plot)?

7. Count reads per genes using HTSeq

Select the **BAM** file and the tool **RNA-seq / Count aligned reads per genes with HTSeq**. Set the parameter **Is the data stranded and how** = "**yes" in HTSeq**.
-Inspect the result files. Which file contains the read counts per each gene? Can you find genes with counts (note that you can sort the table by clicking on the title of the count column)?
-How many alignments were not counted for any gene (check htseq-count-info.txt)?

8. Save session, get analysis history file, save and run an analysis workflow

Save session: Select **File / Save cloud session**. Give a name to your session and save it.
Save an automatic workflow: Select file **hESC.fastq** and **Workflow / Save starting from selected**.
Run workflow: Select **GM12878.fastq** and **Workflow / Run recent / *yourName*.bsh**.

9. Create count table and description file for the experiment

Select **both tsv files** containing the read counts, and the tool **Utilities / Define NGS experiment**. Set the parameters **Does your data contain genomic coordinates** = **yes** and **Count column** = **count**.
In the resulting **phenodata.tsv** file, fill in the **group** column: enter **1** for hESC and **2** for GM12878. Save as local session.

10. Detect differentially expressed genes with edgeR

Select the file **ngs-data-table.tsv** and run the tool **RNA-seq / Differential expression using edgeR**.
-How many differentially expressed genes are detected (check the number of rows in de-list-edger.tsv)?

11. Annotate the list of differentially expressed genes

Select the file **de-list-edger.tsv** and run the tool **Utilities / Annotate Ensembl identifiers**.
-Which gene has the highest positive fold change?

12. Visualize differentially expressed genes in Chipster genome browser

We will use the file **de-list-edger.bed** as a navigation aid. Visualize it as a **spreadsheet**, and click **Detach** to open it in a separate window. Sort it by fold change (column4) so that the gene with the highest positive fold change is at the top. Put this new window aside for a moment.
Select again **de-list-edger.bed**, both BAM files and the visualization method **Genome browser**. **Maximize** the visualization panel, select **genome = Homo sapiens hg38**, and click **Go**.

-Click on the start position of the first gene in the detached BED file to navigate to that gene. In the **Settings** tab, change coverage scale to 1000. Zoom in and out with a mouse wheel. You can turn off the reads (in the Options, untick the box Reads) and view just the coverage first.
-Does this gene (EEF2) seem to be differentially expressed? Are all the reads located in exons?