RNA-seq data analysis workshop 7.-10.1.2014 / Chipster hands-on tutorial

Eija Korpelainen
chipster@csc.fi

1. Start Chipster and import a fastq file

Go to **http://chipster.csc.fi/**, and **Launch Chipster v2.9**. Log in (username rna2014, password training). Select **File / Open session**, navigate to the Chipster folder and select the file **ENCODE_RNAseq.zip**.

2. Quality control with FastQC

Select file **hESC.fastq**, the tool **Quality control / Read quality with FastQC** and click **Run**. Inspect the results.

-how many reads are there and how long are they? What quality encoding is used?
-is the base quality good all along the reads?

3. Trim sequences based on quality

Select **hESC.fastq** and the tool **Utilities / Trim reads by quality** and set the parameter **Trim 3' end by quality** = **20**. Select the result file **trimmed.fastq** and check the quality again with FastQC. Does the quality look better? What happened to the read number and why?

Run also **Quality control / Read quality with PRINSEQ**. Select the result file **reads-stats.html** and visualization method **Open in external web browser**. How is the read length distribution now?

4. Filter out reads that are shorter than 50 bases.

Select **trimmed.fastq** and the tool **Filtering / Filter reads for length**, and set the parameter **Minimum length** = **50**. How many reads get discarded (check the file **filter.log**)?

5. Align reads to reference genome using TopHat2

Select **accepted.fastq** and run the tool **Alignment / TopHat2 for single end reads**. This takes about 15 min. Right-click on the resulting BAM file and **rename** it to hESC.bam. Rename the index file to hESC.bam.bai. Inspect the file tophat-summary.txt. What was the overall alignment rate? How many reads have multiple alignments?

6. Count reads per genes using HTSeq

Select the **BAM** file and run **RNA-seq / Count aligned reads per genes with HTSeq**. Select the result file **htseq-counts.tsv** and run **Filtering / Filter table by column value** (column = count, cutoff = 1, criteria = larger than) to check how many genes have counts.

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

Save session: Select **File / Save session**. Give a name to your session and save it.
Get a textual report: Select **filtered-NGS-result.tsv** and click on the small paper icon in the workflow panel.
Save an automatic workflow: Select file **hESC.fastq** and **Workflow / Save starting from selected**.
Run workflow: Select file **GM12878.fastq** and in the upper panel select **Workflow / Run recent / yourName**

8. Create count table and description file for the experiment

Select **both htseq-counts.tsv** files 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 file, fill in the **group** column: enter 1 for hESC and 2 for GM12878. Save the session.

9. Detect differentially expressed genes with DESeq

Select the file **ngs-data-table.tsv** and run the tool **RNA-seq / Differential expression using DESeq** so that you change the parameter **Dispersion estimation method** = **local**.

Are the differentially expressed genes found reliable? Hint: click **More help** to check how dispersion is calculated when you don't have replicates.

10. Visualize differentially expressed genes in genome browser

Open **de-list-deseq.bed,** click **Detach** and put the new window aside. Select the BED file, both BAM files and visualization method Genome browser. **Maximize** the visualization panel, select **genome Human hg19**, and click **Go**. Click on the rows of the detached BED file to navigate from one differentially expressed gene to another. Change coverage scale to 100. Zoom in and out with a mouse wheel.

11. Open new session

Select **File / Open** session and the file **pasilla.zip**. Inspect the phenodata. This is a two-group comparison with 7 samples. Some samples were sequenced single end (readtype = 1), some paired end (readtype = 2).

12. Analyze differential expression with edgeR classic

Select the file **pasilla_counts.tsv** and run the tool **RNA-seq / Differential expression using edgeR**.

-how many differentially expressed genes do you get?
-is common dispersion a good approximation of genewise dispersions, as judged by the dispersion plot?
-how big fold change is required for differential expression, as judged by the MA plot?
-do the groups separate along the dimension 1 in the MDS plot?

13. Analyze differential expression with DESeq

Select the file **pasilla_counts.tsv** and run **RNA-seq / Differential expression using DESeq**.

-how many differentially expressed genes do you get?

14. Compare the result lists from DESeq and edgeR using Venn diagram

Select the files **de-list-deseq.tsv** and **de-list-edger.tsv** by keeping the ctrl key down. In the visualization panel pull-down menu select the method **Venn-diagram**. Select genes found by both methods and create a new dataset out of them (go to the tab "selected" and click the button "create new dataset").

15. Analyze differential expression with edgeR glm

Select the file **pasilla_counts.tsv** and run the tool **RNA-seq / Differential expression using edgeR for multivariate experiments** so that you analyze only genes which are **expressed in at least 3 samples**.

Check how many genes have a p-value < 0.05: Select the file **edger-glm.tsv** and tool **Filtering / Filter table by column value** and change the parameters:

-**Column to filter by** = **PValue-as.factor(group)2**

-**Does the first column have a title** = **no**

Repeat the analysis so that you take into account the fact that some samples were sequenced single end, some paired end: Set **Main effect 2 = readtype** (and analyze only genes which are **expressed in at least 3 samples**). Filter the result file as above. How many differentially expressed genes do you get now?

Save session.