# Euler cluster hands-on exercises: RNA-seq primary analysis

The goal of the exercise is to run a full primary analysis of a single file from Illumina RNA-seq experiment. The primary analysis includes mapping to the reference genome, conversion from SAM to BAM alignment formats, sorting and indexing the alignment files, counting of reads in the genes.

## Preparation

Log in to Euler.

Create the folder to run the primary analysis jobs in the personal scratch

cd /cluster/scratch/yourname

mkdir Euler\_test

cd Euler\_test/

Copy a single fastq.gz file and the template of bash script template.sh to your new folder in scratch

cp /cluster/scratch/michalo/data/mfc\_dac\_1.fastq.gz \ /cluster/scratch/yourname/Euler\_test

cp /cluster/scratch/michalo/Euler\_test/template.sh \

/cluster/scratch/yourname/Euler\_test

## STAR alignment

Copy the template into a script for STAR

cp /cluster/scratch/michalo/template.sh star.sh

Fill the template. Module to call: star

module load stack/.2024-06-silent gcc/12.2.0

module load star/2.7.10b

Command line in the bash:

STAR --genomeDir /cluster/home/michalo/scratch/genomes \

--readFilesIn my.fastq.gz --readFilesCommand zcat \

--runThreadN 16 --genomeLoad LoadAndKeep \

--outFileNamePrefix my

Command line of the job to run:

chmod a+x star.sh

sbatch -n 1 --cpus-per-task=16 --time=3:00:00 \

--mem-per-cpu=4096 \

--wrap="/cluster/home/michalo/scratch/Euler\_test/star.sh"

Run the job and try to see the execution with “squeue” command.

## Samtools conversion and sorting

Copy the template to a script for samtools.

cp template.sh samtools.sh

Fill the template. Module to call: samtools

module load stack/.2024-06-silent gcc/12.2.0

module load samtools/1.16.1

Command line in the bash:

samtools view -bS -@ 16 myAligned.out.sam > my.bam

samtools sort -T my -@ 16 -O bam my.bam > my.sorted.bam

Command line of the job to run

sbatch -n 1 --cpus-per-task=16 --time=3:00:00 \

--mem-per-cpu=4096 \

--wrap="/cluster/home/michalo/scratch/Euler\_test/samtools.sh"

## Samtools index

This is efficient with a single core, can be run in the command line of a login node, or in a job with --cpus-per-task=1

samtools index my.sorted.bam

## Counting in genes

Copy the template to a script for featureCount

cp template.sh count.sh

Fill the template. Module to call: subread

module load stack/.2024-06-silent gcc/12.2.0

module load subread/2.0.6

Command line in the bash:

featureCounts -M -s 0 -T 16 -t gene -g gene\_id \

-a /cluster/home/michalo/scratch/genomes/Homo\_sapiens.GRCh38.111.gtf \

-o counts.txt my.sorted.bam

Command line of the job to run

sbatch -n 1 --cpus-per-task=16 --time=3:00:00 \

--mem-per-cpu=1096 \

--wrap="/cluster/home/michalo/scratch/Euler\_test/count.sh"

## Final tasks and questions:

* Check the resource footprint of your jobs in:
  + <https://slurm-jobs-webgui.euler.hpc.ethz.ch/>
* What are the sizes of aligned data in SAM, BAM and sorted BAM? Why such a difference?
* Check the statistics of alignment (myLog.final.out).
  + What is the percentage of uniquely and multiply mapped reads?
  + Can you confirm that the sequencing data are human?
* Check the statistics of counting (counts.txt.summary)
  + Any potential problems or surprises?
* Check the count table
  + Are the genes expressed as you expected?
  + What is the number of reads aligned to of RCC2?