
CHAPTER
THREE

PROJECT

3.1 Introduction

The goal is to reproduce parts of the analysis described in this paper (to read):

- <https://www.nature.com/articles/s41467-020-15966-7>

We want to analyze the RNA-Seq data in order to find **differentially expressed genes**, i.e. genes that are more (or less) expressed in one condition (persisters) compared to another (control).

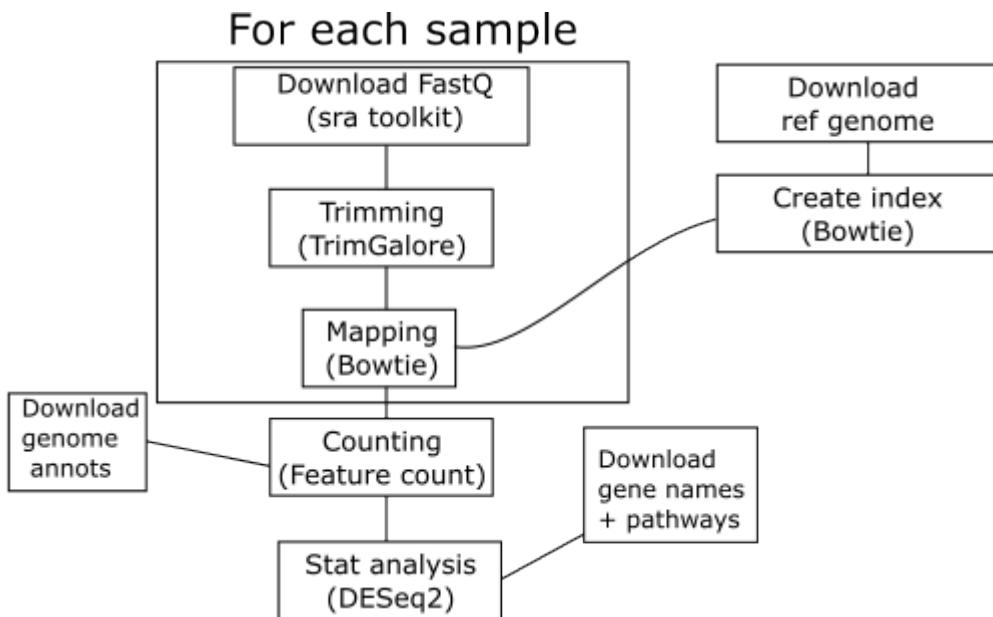
To do so you will design and implement a workflow (Snakemake or Nextflow) that must be reproducible. The work will be done in small groups (4 students).

1. This workflow must use containers that you will *build yourself* (Docker or Apptainer) to run the processes;
2. The code must be readable, commented, and documented;
3. We should be able to re-execute easily the workflow;

Deliverables:

1. Container recipes (Dockerfiles, Apptainer recipes)
2. Workflow code (Nextflow files + configuration or Snakemake files)
3. README.md + run.sh with all instructions for us to reproduce *easily* your analysis
4. **Report with the following parts:**
 - a. Introduction (Reproducibility, biological topic of the papers to reproduce, etc.)
 - b. Material and Methods (Tools used, and setup)
 - c. Results (Workflow developed, results obtained after execution)
 - d. Conclusion / perspectives (Interpretation of the results, and conclusion about reproducibility)
5. Oral presentation: Each group will make a presentation (**December 12th?**).
6. **Mid-project evaluation / progress report**
[November 7th:]
 - a. 10 minutes oral presentation
 - b. **First version of the code (in the git repository) with:**
 - All the image recipes (working) and
 - First steps of the workflow (data download, genome download and indexing) (working)

3.2 Workflow to implement and execute



3.3 Downloading FASTQ files

- Tool: `fastq dump`
- Commands:

```
$ fasterq-dump --threads <#CPUS> --progress <SRAID>
$ gzip *.fastq
```

If you find better / quicker ways of downloading the data, feel free!

3.4 Trimming the reads

- Commands:

```
$ trim_galore -q 20 --phred33 --length 25 <FASTQ FILE>
```

3.5 Downloading reference genome

- Commands:

```
$ wget -q -O reference.fasta "https://eutils.ncbi.nlm.nih.gov/entrez/eutils/efetch.fcgi?
db=nuccore&id=CP000253.1&rettype=fasta"
```

3.6 Downloading reference genome annotations

- Commands:

```
$ wget -O reference.gff "https://www.ncbi.nlm.nih.gov/sviewer/viewer.cgi?db=nuccore&report=gff3&id=CP000253.1"
```

3.7 Creating genome index

- Tool: Bowtie
- Commands:

```
$ bowtie-build <full genome fasta file> <index name>
```

3.8 Mapping FastQ files

- Tool: FastQC
- Commands:

```
$ bowtie -p <#cpus> -S <INDEX NAME> <(gunzip -c <GZIPED FASTQ FILE>) | \
    samtools sort -@ <#CPUS> > <NAME>.bam
$ samtools index <NAME>.bam
```

3.9 Counting reads

- Tool: subread
- Commands:

```
$ featureCounts --extraAttributes Name -t gene -g ID -F GTF -T <#CPUS> -a <GFF> -o \
  <counts.txt <BAM FILES>
```

With options: #. **-t gene** indicates that counts should be done on gene. You may use other features such as **exon** #. **-g ID** selects the gene ID to store in the output file. This depends on your input GFF file .

3.10 Statistical analysis (differential gene expression)

- Tool: DESeq2
- Commands: You will have to find the method and write the script yourself.

3.11 Additional informations

- Some steps are not highly described, for example:
 - Getting the mapping between gene identifiers (in the GFF file) and gene names (needed in the final MA plot). This can be downloaded from [AureoWiki](#).
 - Getting the list of genes involved in translation. This can be found in [KEGG](#), using the REST api for example.