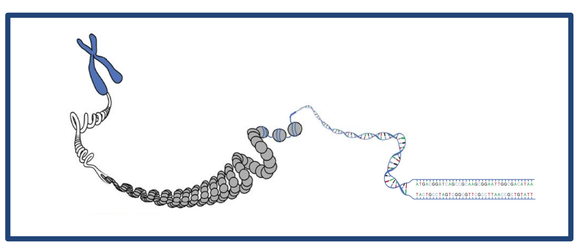
Gene-regulation project: high-throughput approaches for robust NGS data analyses

Claire Rioualen, Jacques van Helden

Theory and Approaches of Genomic Complexity (TAGC), INSERM U1090, Aix-Marseille Université, Marseille F-13288, France

Objectives & Approach

Next generation sequencing (NGS) has gotten faster and cheaper in the past few years. What we are lacking now are means to analyse the incredibly huge amount of data generated.



We have decided to handle this question. Our main focuses include:

- **reproducibility** of the biological analyses;
- **traceability** of the tools and parameters used;
- **flexibility** of the framework;
- **portability** in different operating systems.

Snakemake[1], a flexible environment for workflow development

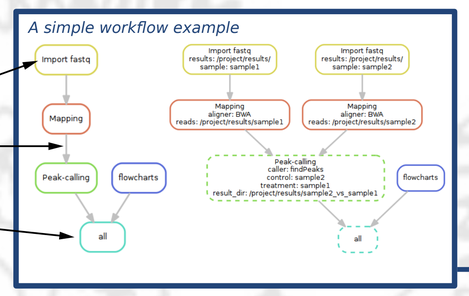
Python library, **GNU make** concepts

Rules are recipes to perform an operation (python, R or shell)

Dependencies between rules inferred by **inputs** and **outputs**

Final **target** defines operations to be performed

1. Köster, Johannes and Rahmann, Sven. "Snakemake - A scalable bioinformatics workflow engine". Bioinformatics 2012.




```
## workflow.py
SAMPLES = ["sample1", "sample2"]
CHIP = "sample2"
CONTROL = "sample1"

rule all:
    input:
        expand("{sample}.bam", sample = SAMPLES)
        expand("{chip}_vs_{control}",
                chip = CHIP, control = CONTROL)
rule mapping:
    input: "{file}.fastq"
    output: "{file}.bam"
    shell: "bowtie {input} > {output}"

rule peak_calling:
    input: chip = "{chip}.bam", control = "{chip}.bam"
    output: "{file}.bam"
    shell: "macs2 {input.chip} {input.control} > {output}"
```

Virtual environments

Virtualization allows running analyses in a **controlled environment**, ensuring **reproducibility** and **portability**.
Several environments were developed, and **tutorials** written accordingly.

A simple framework

Data description

```
; Sample description file
; Organism: Saccharomyces cerevisiae
; Experiment type: ChIP-seq
;
ID	Condition	ENA_experiment	Scan Name
GSM521934	input	SRX021359	SRR051930
GSM521935	chip	SRX021358	SRR051929
```

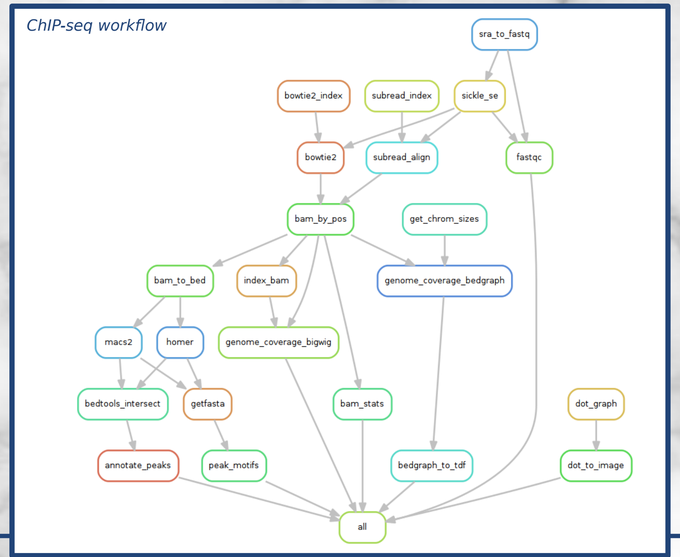
```
; Design for the peak-calling of ChIP-seq
treatment	control	description
GSM521935	GSM521934	TBF1_occupancy
```

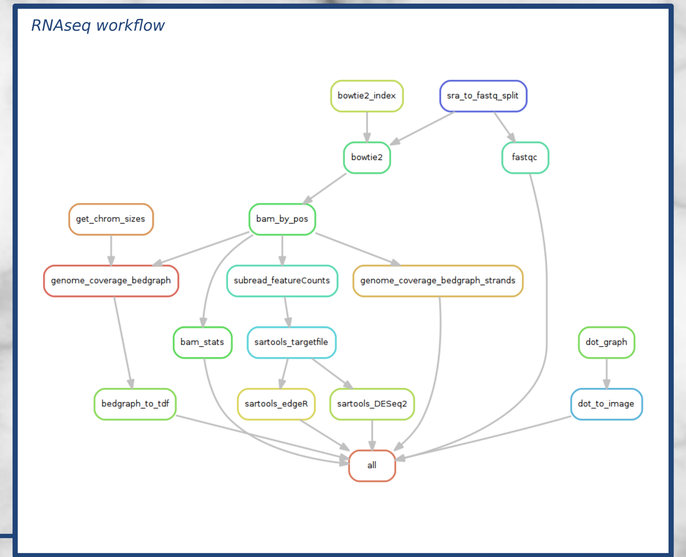
Config of the analysis

```
metadata:
  samples: "metadata/samples.tab"
  design: "metadata/design.tab"
  configfile: "metadata/config.yml"
  seq_type: "se"

dir:
  fastq: "fastq"
  genome: "genome"
  results: "results"

tools:
  trimming: "sickle"
  mapping: "bowtie2"
  peakcalling: "macs2 homer macs14 spp swembl"

bowtie2:
  threads: "5"
  max_mismatches: "1"
```

Workflow design

```
include: config["gene_regulation"]

# Samples
SAMPLE_IDS = read_table(config["metadata"]["samples"])['ID']
TREATMENT = read_table(config["metadata"]["design"])['treatment']
CONTROL = read_table(config["metadata"]["design"])['control']

# Rules
include: "scripts/snakefiles/rules/bowtie2.rules"
include: "scripts/snakefiles/rules/fastqc.rules"
include: "scripts/snakefiles/rules/macs2.rules"
include: "scripts/snakefiles/rules/homer.rules"

# Targets
FLOWCHARTS      = expand("reports/{graph}.png", graph=["dag", "rulegraph"])
IMPORT          = expand("fastq/{samples}/{samples}.fastq", samples=SAMPLE_IDS)
QUALITY_CONTROL = expand("fastq/{samples}/{samples}_fastqc/{samples}_fastqc.html", samples=SAMPLE_IDS)
MAPPING         = expand("samples/{samples}/{samples}.bam", samples=SAMPLE_IDS)
PEAK_CALL       = expand(expand("peaks/{{peakcaller}}/{treat}_vs_{control}.bed",
                                zip, treat=TREATMENT, control=CONTROL), peakcaller=["macs2","homer"])

# Workflow execution
rule all:
    """Run all the required analyses."""
    input: FLOWCHARTS, IMPORT, QUALITY_CONTROL, MAPPING, PEAK_CALL
    params: qsub = config["qsub"]
```



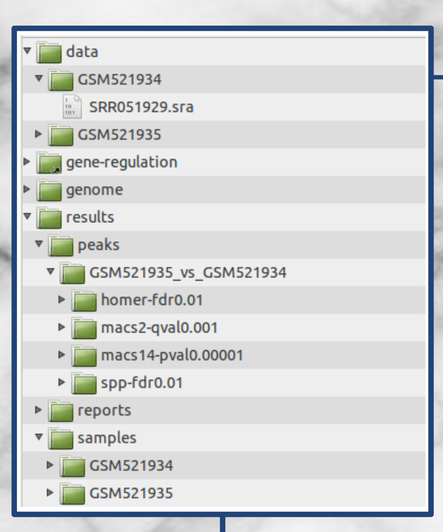


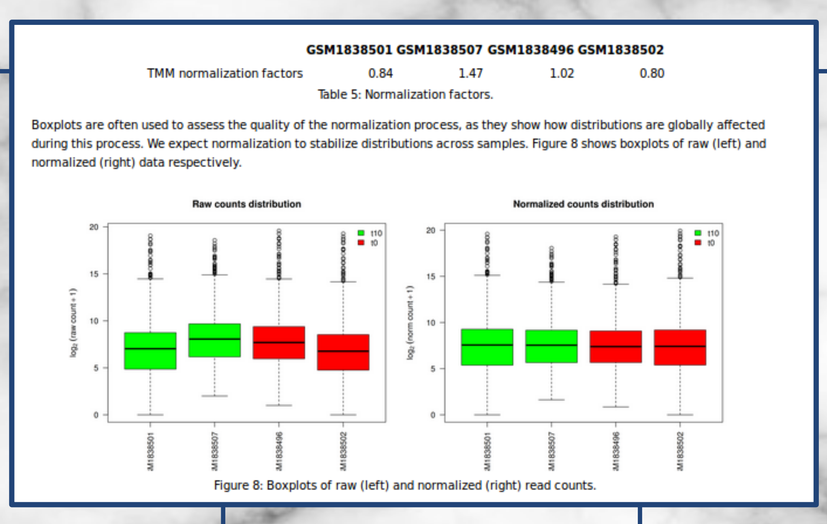
From a transparent data organization...

...to an exhaustive & reproducible analysis

A variety of visualization tools

Intuitive file organization provides accessibility for biologists and bioinformaticians





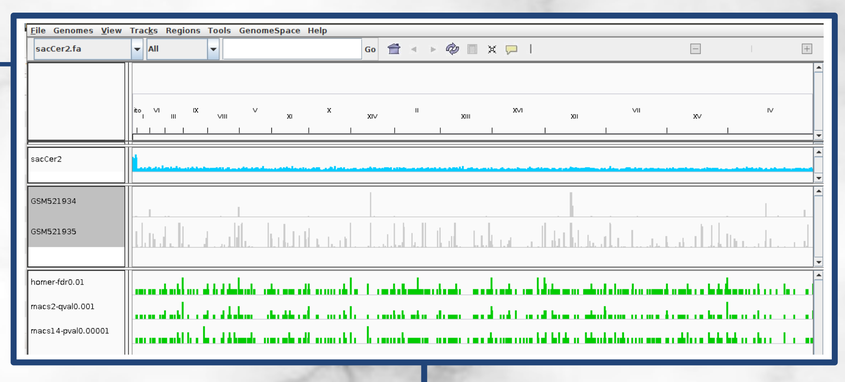
GSM1838501 GSM1838507 GSM1838496 GSM1838502

TMM normalization factors 0.84 1.47 1.02 0.80

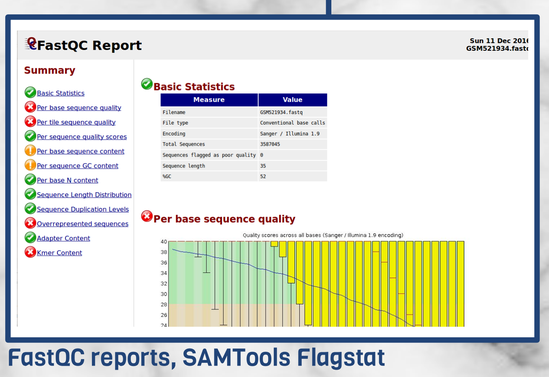
Table 5: Normalization factors.

Boxplots are often used to assess the quality of the normalization process, as they show how distributions are globally affected during this process. We expect normalization to stabilize distributions across samples. Figure 8 shows boxplots of raw (left) and normalized (right) data respectively.

Figure 8: Boxplots of raw (left) and normalized (right) read counts.



A view of the ChIP-seq peaks detected by the workflow, under the Integrative Genome Browser (IGV)



FastQC reports, SAMTools Flagstat

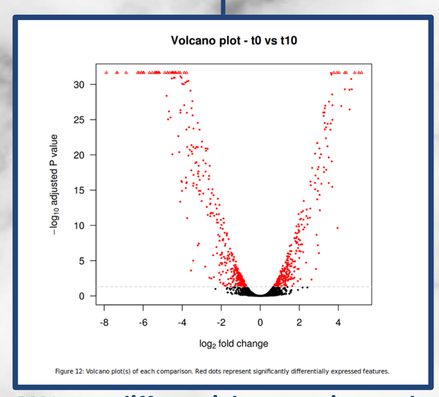


Figure 12: Volcano plot(s) of each comparison. Red dots represent significantly differentially expressed features.

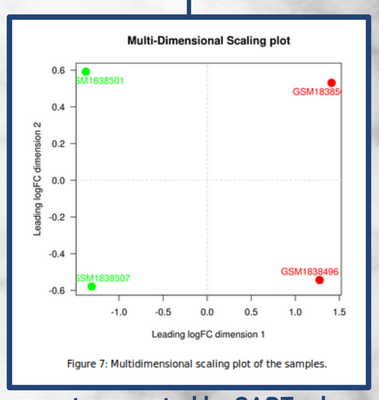


Figure 7: Multidimensional scaling plot of the samples.

RNA-seq differential expression analysis report generated by SARTools

Quality control

Normalization and significativity of the results

Visualization in genomic context

Discussion & Perspectives

We have developped a set of **reusable** workflows and tools for NGS analysis. For this purpose we chose to use the **Snakemake** workflow engine, which was able to fulfill our expectations:

Modularity: one can pick up any set of rules in order to build a custom pipeline.

Recycling: the rules can be used in several workflows, by several people.

Benchmarking: a number of alternative software are already available for benchmarking the peak-calling and the mapping steps.

Flexibility: config file allows tuning parameters to cater workflows to different species (bacteria, plants, mammalians...) or conditions.

Collaboration: several developpers can share their rules & workflows.

Portability: pipelines can be run on servers, PCs and virtual environments.

Accessibility: all the material is versioned & available in a public GitHub repository.

The number of Snakemake users is growing in France, which allowed us to create a **community** called "**CoBRAS**" (Communauté de Bioinformaticiens Rassemblés Autour de Snakemake), and to organize meetings in order to exchange methods and good practices.

Our work is available under several platforms, including GitHub and the Docker Hub. It was also used in a **protocol published** recently.

Upcoming developments include a **collaboration** with the **University of Mexico**, funded by the **NIH**, which purpose is to integrate public NGS data to enrich the RegulonDB database on *E. coli* regulation.

Acknowledgments: This work is funded by France Génomique, NIH grant GM0110597 & FOINS-CONACYT - Fronteras de la Ciencia 2015 - ID 15. I would like to thank Lucie Khamvongsa, Jocelyn Brayet, Bruno Contreras-Moreira, Julio Collado-Vides & all his team for their great contributions to this work.