BraunLabPipeline

Pipeline for High-throughput sequencing data.

Table of contents

General information

This document contains all information to access and run the pipeline used in the Braun lab for processing ChIP-seq, ATAC-seq and RNA-seq data. The goal of this pipeline is to provide an automated solution to generate sequencing files (fastq), mapping (bam), peak and gene quantification.

Please note that this pipeline has been set to run on UNIGE cluster (baobab) and modifications may be required to run it on another cluster.

What is inside the pipeline

The pipeline starts with sequence files (fastq.gz) and uses cutadapt for trimming reads if necessary. For mapping, it uses either bowtie2 (ChIP-seq / ATAC-seq) or STAR (RNA-seq). Picard-Tools is used for marking duplicated reads. SamTools is used for sorting, indexing and filtering reads. For peak calling we use MACS2 and gene quantification is performed using featureCounts.

You can run all tasks into one go or run specific tasks depending on your needs.

####### Contact

If you need more information: Nikolaos Lykoskoufis: nikolaos.lykoskoufis@unige.ch

Steps of the pipeline

Task list

The pipeline can be used to do many things. You can run each task separately (only run filtering reads for example) or you can run the whole thing in one go. Calling task "all" will run all task. Here is the list of tasks you can run and what they each do:

Task	Description
1	Trimming of reads
1.1	QC of fastq files
2	Mapping reads
3	Marking PCR duplicates **
4	Filtering reads **
4.1	Fragments size distribution QC *
4.2	QC of bam files
5	Bam to BW
6	Bam to Bed **
7	Read extension **
8	Peak calling **
8.1	Peak counts **
9	Gene quantification ***

* Exclusive for ATAC-seq data

** Exclusive for ATAC-seq and ChIP-seq data

*** Exclusive for RNA-seq data

Command line

Each task require some arguments. Here is the exhaustive list of arguments you can provide for each task.

Argument	Definition
-cf	Absolute path to the configuration file
-raw	Absolute path to the raw directory *
-od	Absolute path to the output directory
-fastq	Abolute path to the fastq directory
-bam	Absolute path to the bam directory
-eqd	Absolute path to the quantification directory
-bed	Absolute path to the bed directory

Argument	Definition
-bw	Absolute path to the bigwid directory
-t	Tasks to be ran. It can be an integer from 1 to 8 or "all" which specifies run all steps

The arguments in bold are required for the pipeline no matter whether you run all the steps or specific steps.

* The raw directory can be used as the output directory as well. If you run the whole pipeline (all tasks), the raw directory is where all subfolders for each task will be generated. Be careful that the specified does not contain subfolders named similarly as the ones generated from the pipeline. For example, if you run task 2 (mapping), the pipeline will try to create a subfolder "bam" inside the raw directory. If another "bam" subfolder is already present then the pipeline will through an error.

If you want to run tasks separately, below you will find the required arguments to pass on the command line.

Task	options required
all	-raw -fastq -cf -t
1	-raw -fastq -cf -t
1.1	-raw -fastq -cf -t
2	-raw -fastq -cf -t
3	-raw -bam -cf -t
4	-raw -bam -cf -t
4.1	-raw -bam -cf -t
4.2	-raw -bam -cf -t
5	-raw -bam -cf -t
6	-raw -bam -cf -t
7	-raw -bed -bam -cf -t
8	-raw -bed -bam -cf -t
8.1	-raw -peak -cf -t
9	-raw -bam -cf -t

This table summarizes mandatory options for each task if there are run separately. All task require the configuration fil (-cf), -t and -raw options.

Output

Each task creates files which are written in specific directories as can be seen in the picture below.

```
raw_dir
├── ATACseqQC
│   └── log
├── bam
│   └── log
├── bed
│   └── log
├── bigwig
│   └── log
├── extended_bed
│   └── log
├── fastq
├── fastQC
│   ├── log
│   └── multiqc_data
├── filtered_bam
│   └── log
├── log
├── marked_bam
│   └── log
├── peaks
│   └── log
├── report
│   └── log
├── sorted_bam
│   └── log
└── trimmed_fastq
    └── log
```

Configuration file

The **configuration file** contains all the necessary paths to software and all the parameters you want to use for each task. Below is an snippet of the configuration file.

```


#### ANNOTATION FILES #####
#annotation,
/srv/beegfs/scratch/groups/funpopgen/data/annotations/gencode.v19.annotation.nochr.gtf
annotation,/srv/beegfs/scratch/shares/brauns_lab/data/annotations/mus_musculus/Mus_musculus.GRCm38.102.modified.txt
###########################

#### PAIRED-END or NOT ####
#Are reads Pair-end read: 1 if they are pair-end, 0 if they are single-end
pairend,0
############################

# Are you mapping ATAC-seq, ChIP-seq or RNAseq data?
technology,RNAseq
```

```
#### READ TRIMMING OPTIONS ####
#Example of how a cutadapt commands looks like
#cutadapt -a CTGTCTCTTATACACATCTCCGAGCCCACGAGAC -A
CTGTCTCTTATACACATCTGACGCTGCCGACGA -m 20 -O 5 -o
testFile_R1_001.trim.fastq.gz -p testFile_R2_001.trim.fastq.gz
testFile_R1_001.fastq.gz testFile_R2_001.fastq.gz
#If you want to modify the parameters used, you can comment the line
below, and add what you want.
trim_reads, -a CTGTCTCTTATACACATCTCCGAGCCCACGAGAC -A
CTGTCTCTTATACACATCTGACGCTGCCGACGA -m 20 -O 5
################################

#### MAPPER ####
#Choose your mapper (possible options: STAR, bwa, bowtie,HiSat2, etc...)
mapper,STAR
#mapper,bowtie2
################
```

All lines starting with a **#** are not read by the pipeline.

The lines that the pipeline are read need to be specially formatted. They should always start with a keyword, as follows: **keyword, parameters to use**

Keywords are essential in the configuration file because they are recognised by the pipeline as is.

Prepare your data

Fastq files

Fastq files should be written in this specific way:

SampleID*.R1_.*fastq.gz and if paired-end data: SampleID*.R2_*.fastq.gz

If you use ChIP-seq, the input files should be written as follows:

Input_SampleID*.R1_*.fastq.gz

If you do not respect the naming convention, the pipeline will FAIL

All other files should start with SampleID.whatever and Input_SampleID.whatever (if ChIP-seq data).

Running the pipeline

You have the option to run all tasks of the pipeline or specify which steps you want to run. If you run the whole pipeline in one go, then you need to specify the --task; --raw-dir; --fastq-dir; -cf ; parameters.

Otherwise, depending on the tasks you decide to run, you will need to use different combination of parameters. For example to convert bam files to bed files you will need to use --task; --raw-dir; --bam-dir; -cf. Or if you want to run task 8 (peak calling), then you need to use --task; --raw-dir; --bed-dir; -cf.

```
python3 main.py --help
#The following command outputs how to use the script.
usage: main.py [-h] [-v] -raw RAW_DIR [-fastq FASTQ_DIR] [-bam BAM_DIR]
               [-peak PEAKS_DIR] [-eqd EQ_DIR] [-bed BED_DIR] [-bw
BIGWIG_DIR]
               [-od OUTPUT_DIR] -cf CONFIG_FILE_PATH -t TASK [TASK ...]

BraunPipeline
 *  Authors     : Nikolaos Lykoskoufis / Simon Braun
 *  Contact     : nikolaos.lykoskoufis@unige.ch / simon.braun@unige.ch
 *  Webpage     : https://github.com/NLykoskoufis/braunATACpipeline
 *  Version     : 1.0
 *  Description : Pipeline to process High throughput sequencing data.

optional arguments:
  -h, --help            show this help message and exit
  -v                    Display pipeline version
  -raw RAW_DIR, --raw-dir RAW_DIR
                        Absolute path to the raw directory
  -fastq FASTQ_DIR, --fastq-dir FASTQ_DIR
                        Absolut path fastq to diretor(y)ies. If multiple
directories, separate eache path with space
  -bam BAM_DIR, --bam-dir BAM_DIR
                        Path bam diretory, is multiple, separate with
space.
  -peak PEAKS_DIR, --peak-dir PEAKS_DIR
                        Path peak diretory, is multiple, separate with
space.
  -eqd EQ_DIR, --quant-dir EQ_DIR, -quantification_dir EQ_DIR
                        Absolut path quantifications diretory
  -bed BED_DIR, --bed-dir BED_DIR
                        Absolut path of where to save/read bed files
  -bw BIGWIG_DIR, --bigwig-dir BIGWIG_DIR
                        Absolut path peak calling diretory
  -od OUTPUT_DIR, --output-dir OUTPUT_DIR
                        Path to output directory. Use it only if you do
not run the pipeline from step
  -cf CONFIG_FILE_PATH, --configuration-file CONFIG_FILE_PATH
                        Name of your configuration file:
project_run_config_V1
  -t TASK [TASK ...], --task TASK [TASK ...]
```

The different arguments of the pipeline are:

Argument	Definition
-cf	Absolute path to the configuration file
-raw	Absolute path to the raw directory
-od	Absolute path to the output directory

Argument	Definition
-fastq	Abolute path to the fastq directory
-bam	Absolute path to the bam directory
-eqd	Absolute path to the quantification directory
-bed	Absolute path to the bed directory
-bw	Absolute path to the bigwid directory
-t	Tasks to be ran. It can be an integer from 1 to 8 or "all" which specifies run all steps

The arguments in bold are required for the pipeline no matter whether you run all the steps or specific steps.

Directory tree generated by pipeline

```
raw_dir
├── ATACseqQC
│   └── log
├── bam
│   └── log
├── bed
│   └── log
├── bigwig
│   └── log
├── extended_bed
│   └── log
├── fastq
├── fastQC
│   ├── log
│   └── multiqc_data
├── filtered_bam
│   └── log
├── log
├── marked_bam
│   └── log
├── peaks
│   └── log
├── report
│   └── log
├── sorted_bam
│   └── log
└── trimmed_fastq
    └── log
```