A reproducible Snakemake pipeline to analyse Illumina paired-end data from ChiP-Seq experiments

Jihed Chouaref, Mattijs Bliek, Marc Galland

The Journal of Open Source Software

A reproducible Snakemake pipeline to analyse Illumina paired-end data from ChiP-Seq experiments

Jihed Chouaref[1], **Mattijs Bliek**[1], **and Marc Galland**[1]

1 Swammerdam Institute for Life Sciences, University of Amsterdam

DOI: 10.21105/joss.01465

Software

- Review
- Repository
- Archive

Submitted: 17 May 2019
Published: 06 June 2019

Summary

Chromatin immunoprecipitation followed by high-throughput sequencing (ChIP-seq) is a powerful tool for investigation the genome-wide distribution of DNA binding protein and their modifications. Yet, the computational analysis of Next-Generation Sequencing datasets is still a bottleneck for most of the experimental researchers. Most often, this type of analysis require multiple steps *i.e.* read quality control, mapping to a reference genome, peak calling, annotation and functional enrichment analysis that are performed by various tools *e.g.* fastp (S. Chen, Zhou, Chen, & Gu, 2018), bowtie2 (Langmead & Salzberg, 2012) or samtools (Li et al., 2009) only to name a few. These various tools require different software dependencies and can have different software versions and/or incompatibilities which might impair the analysis reproducibility. Here we provide a complete, user-friendly and highly customized ChIP-seq analysis pipeline for paired-end (Illumina) data based on the Snakemake workflow manager (Koster & Rahmann, 2012).

To make use of the pipeline, only a few modifications are needed. First, software parameters, working and temporary directories as well as genomic references need to be changed in the configuration file (`config.yaml`) that is encoded in the human readable YAML format. Secondly, the user needs to adapt the `units.tsv` tabular file that links sample information to experimental conditions and paired-end fastq files. When these two files are modified, the ChIP-seq pipeline become suitable for any organism from which the genome has been sequenced and annotated. The scalability and reproducibility of the data analysis is ensured by the use of containerization (a Singularity image) and Snakemake through creation and deployment of one virtual environment per rule to manage different software dependencies (*e.g.* Python 2 or 3) using the Conda package manager (https://conda.io) and the Bioconda software distribution channel (Grüning et al., 2018). Raw Illumina paired-end data are processed by the pipeline and are subsequently trimmed, mapped and processed automatically according to the parameters set in the configuration file. A complete Directed Acyclic Graph (DAG) of the different tasks accomplished can be seen in [Figure 1]. If the `singularity` software is available on your machine and you want to use 10 CPUs (`--cores 10`), then run `snakemake --use-conda --use-singularity --cores 10`. Otherwise, run `snakemake --use-conda --cores 10`

The outputs delivered by the pipeline are:
1. Quality controls files to check for the quality of the reads. Reads are processed by programs such as `fastp` and `deeptools` (Ramírez et al., 2016) in order to produce graph that are easily readable and inform quickly about the quality of the experiment. 2. Portable visualization files (bigwig) for the observation of the read coverage on the genome using genome viewer **Figure 2**. 3. Peaks informations files, these `bed` files gather the information about the peak calling produced by the MACS2 algorithm. This files can be potentially used for annotation and functional enrichment analysis. 4. The deeptools suite used by the pipeline produces beautiful visualization of the read coverage

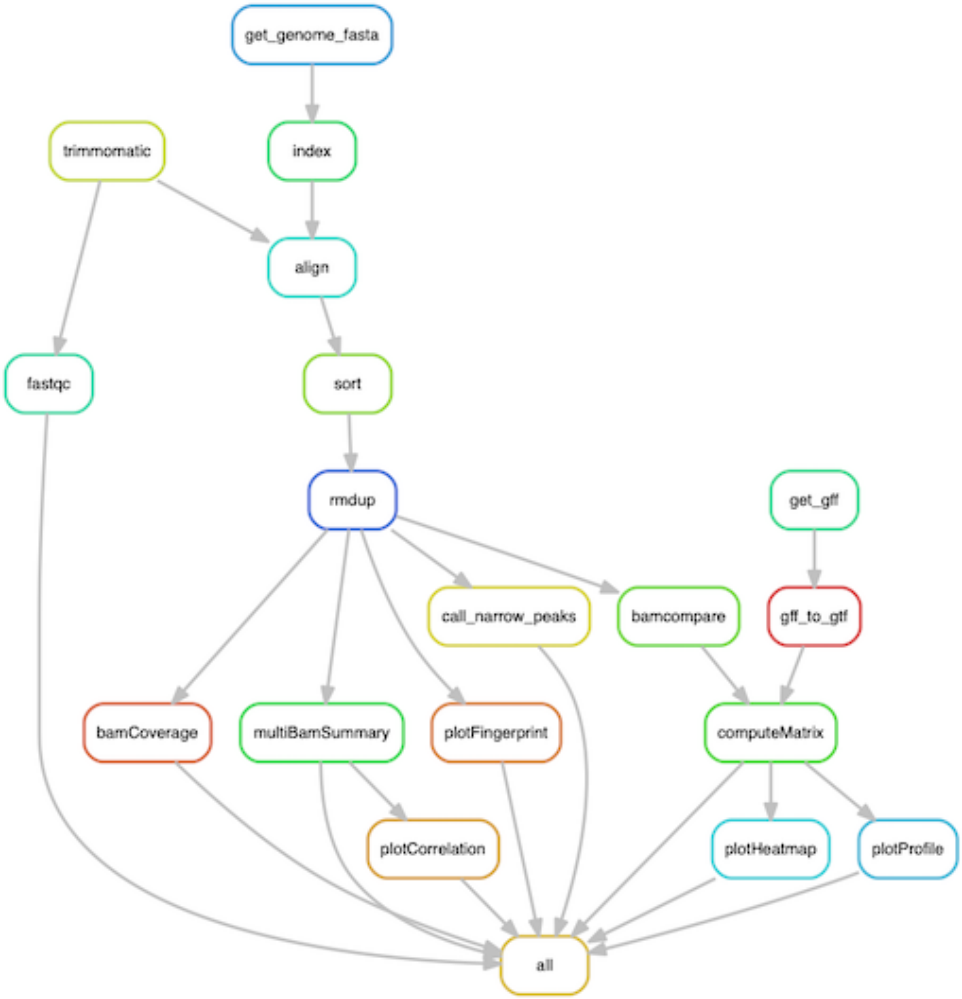
over genomic features provided by the user and a series of quality control tools **Figure 3 and 4**.

This Snakemake ChIP-seq analysis pipeline provides an easy to use command-line pipeline requiring minimum modifications with high modularity for domain knowledge input from the user. The source code of this pipeline has been archived to Zenodo with the following linked DOI doi.141444770 (Chouaref, Galland, & Bliek, 2018).

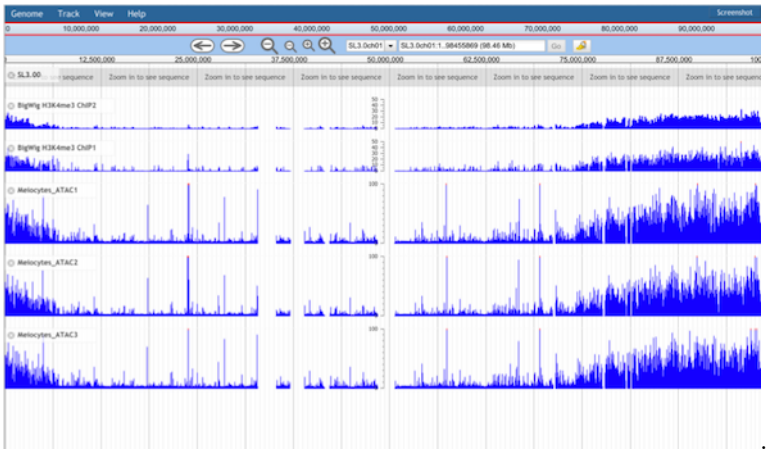
This pipeline also provides a small subsample of sequencing reads generated from random selection of tomato ChIP-seq data, which could be used to quickly modify and test the pipeline to suit specific requirement from the user.

Figures

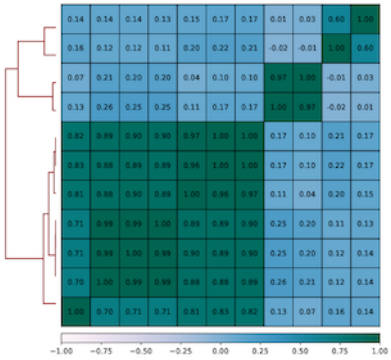
- **Figure 1**: A Directed Acyclic Graph (DAG) of the Snakemake ChIP-seq PE pipeline. This graph has been produced with the command: `snakemake --rulegraph |dot -Tpng > dag.png`.



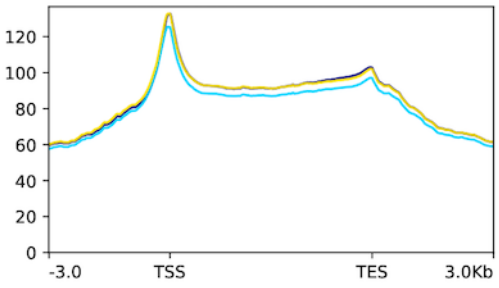
- **Figure 2**: ChIP-seq tracks generated using the pipeline and visualized using JBrowse (Buels et al., 2016).



- **Figure 3**: A pearson correlation plot generated by the pipeline using Deeptools to control for the quality of the experiment.



- **Figure 4**: A profile plot showing the distribution of the reads over a selected genomic feature, here genes are displayed.



Acknowledgements

We acknowledge contributions from Ming Tang and Johannes Köster for their inspired scripts. We would also like to thank the group of RNA biology and applied bioinformatics of the Swammerdam institute for Life sciences for providing the computational ressources, especially Wim de Leeuw and Han Rauwerda. This project has been funded by the People Programme (Marie Curie Actions) of the European Union[1]s Seventh Framework

Programme FP7/2007-2013/ under REA grant agreement n°[606956]13.

References

Buels, R., Yao, E., Diesh, C. M., Hayes, R. D., Munoz-Torres, M., Helt, G., Goodstein, D. M., et al. (2016). JBrowse: A dynamic web platform for genome visualization and analysis. *Genome Biology*, *17*(1), 66. doi:10.1186/s13059-016-0924-1

Chen, S., Zhou, Y., Chen, Y., & Gu, J. (2018). Fastp: An ultra-fast all-in-one FASTQ preprocessor. *Bioinformatics (Oxford, England)*, *34*(17), i884–i890. doi:10.1093/bioinformatics/bty560

Chouaref, J., Galland, M., & Bliek, T. (2018, December). doi:10.5281/zenodo.2025836

Grüning, B., Dale, R., Sjödin, A., Chapman, B. A., Rowe, J., Tomkins-Tinch, C. H., Valieris, R., et al. (2018). Bioconda: Sustainable and comprehensive software distribution for the life sciences. *Nature Methods*, *15*(7), 475–476. doi:10.1038/s41592-018-0046-7

Koster, J., & Rahmann, S. (2012). Snakemake–a scalable bioinformatics workflow engine. *Bioinformatics*, *28*(19), 2520–2522. doi:10.1093/bioinformatics/bts480

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. *Nature Methods*, *9*(4), 357–359. doi:10.1038/nmeth.1923

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., et al. (2009). The Sequence Alignment/Map format and SAMtools. *Bioinformatics (Oxford, England)*, *25*(16), 2078–2079. doi:10.1093/bioinformatics/btp352

Ramírez, F., Ryan, D. P., Grüning, B., Bhardwaj, V., Kilpert, F., Richter, A. S., Heyne, S., et al. (2016). deepTools2: A next generation web server for deep-sequencing data analysis. *Nucleic Acids Research*, *44*(W1), W160–165. doi:10.1093/nar/gkw257