Developing and sharing reproducible bioinformatics pipelines: best practices

Yohann LELIEVRE[1], Audrey BIHOUÉE[2], Eric CHARPENTIER[2], Alban GAIGNARD[2,4], Simon SOUCHET[3] and Damien VINTACHE[1]

[1] LS2N, UMR CNRS 6004, IMT Atlantique, ECN, Université de Nantes, Nantes, France.
[2] l'institut du thorax, INSERM, CNRS, Université de Nantes, Nantes, France.
[3] Angers Academic Hospital, CHU d'Angers, France
[4] Nantes Academic Hospital, CHU de Nantes, France

Corresponding author: Yohann.Lelievre@univ-nantes.fr

1 Introduction

Life-sciences are nowadays conducted in multi-disciplinary and multi-centric studies. In this context, the same software components must be deployed in multiple environments for reproducibility and scalability issues. In addition, data analysis pipelines are usually composed of multiple components, continuously evolving, which leads to maintenance and long-term support challenges. To promote FAIR [1] principles, providing controlled software environments becomes mandatory. We propose a set of best practices taking advantage of proven or promising tools: Git, Conda, SnakeMake[1], Jenkins and Docker.

2 Motivations

Bio-informaticians and software developers need to build data analysis pipelines in controlled environments to ensure long-term re-execution and better reproducibility. From an end-user point of view, typically a biologist, data analysis pipelines should be automatically installable in a local or dedicated computing infrastructure, including any software or data dependency. Pipelines should be launched in three steps: i) environment setup/activation, ii) parameters tuning, and iii) pipeline execution.

3 Approach and Results

The BiRD pipeline registry results from applying these guidelines in the context of Exome sequencing and RNAseq (variant calling, differential gene expression, gene fusion detection, single-cell). These pipelines are described in a GitLab web portal. GitLab allows i) to document the pipeline an its usage, and ii) to host and version the associated source code. To ease installation and dependency management, we packaged and deployed the executable software components through the Conda package manager in a dedicated repository [2]. To assess their long-term re-execution, workflows and associated software environments are nightly assembled into minimal Docker images through a Jenkins continuous integration system.

4 Conclusion and perspectives

The best practices hereby proposed aim at promoting findable and accessible data analysis pipelines through web-based resources. This process allows to re-package and re-execute pipelines in the long run, and to adapt to continuously evolving environments. Our future works include two main directions: i) handling data resources as part of the pipeline distribution process (*e.g.* BioMaj), and ii) studying how to promote interoperability between multiple systems and infrastructures. To enhance trust for end-users and to encourage reuse, provenance metadata and controlled vocabularies (*e.g.* EDAM) offer interesting perspectives to associate produced/analyzed with large-scale bio-resource registries such as BioTools.

Acknowledgements

This work was supported by the BiRD core facility, the SyMeTRIC project and the GRIOTE project.

References

[1] Johannes Köster and Sven Rahmann. Snakemake - a scalable bioinformatics workflow engine. *Bioinformatics*, (28 (19)):2520–2522, 2012.

1. Findable - Accessible - Interoperable - Reusable
2. https://anaconda.org/BiRD