Developing and sharing reproducible bioinformatics pipelines: best practices

Y. Lelièvre¹, A. Bihouée², E. Charpentier², A. Gaignard²,4, S. Souchet³ and D. Vintache¹

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

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 (Findable – Accessible – Interoperable – Reusable) principles, providing controlled software environments becomes mandatory. We propose a set of best practices taking advantage of proven or promising tools: Git, Conda, SnakeMake, Jenkins and Docker.

Developers

Build reproducible pipelines

Developers

End-users

Deploy and launch pipelines

Developers

End-user

Results

Achieved pipelines:
☑ SingleCell RNASeq
☑ Exome variant calling

In progress pipelines:
☐ DGESeq
☐ Gene fusion detection
☐ RNASeq variant calling
☐ RNASeq Differential gene expression

Web portal:
http://bird_pipeline_registry.univ-nantes.io/PipelinesPortal

Discussion

Benefits:
F: Indexed and searchable packages on https://anaconda.org
A: Web-based package management and installation
    Controlled deployment on Linux containers / systems
I: Virtual environments to handle incompatible libraries
    Multi-platform, multi-language: Snakemake + Conda
R: Versioned software environments to foster reproducibility

Limitations:
○ Heavy data resources required (reference genomes, etc.)
○ Some tools / libraries need to be packaged beforehand

Conclusion

The best practices hereby proposed aim at promoting findable and accessible data analysis pipelines through web-based resources. This process allows to 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.