**[Name]: A Container-based Reproducible Untargeted Metabolomics Data Processing Pipeline**

Xinsong Du1, Luran Manfio1, Alexander Kirpich2, William Hogan1, Timothy J. Garrett3, Dominick J. Lemas1

1 Department of Health Outcomes and Biomedical Informatics, College of Medicine, University of Florida

2 Department of Population Health Sciences, School of Public Health, Georgia State University

3 Department of Pathology, Immunology and Laboratory Medicine, College of Medicine, University of Florida

Keywords: febrile neutropenia; machine learning; mortality; cancer; HCUP

Author to whom correspondence should be addressed:

Dominick J. Lemas, Ph.D

Assistant Professor

Department of Health Outcomes and Biomedical Informatics

University of Florida College of Medicine

2004 Mowry Road- Clinical and Translational Research Building

Gainesville, FL 32608

Ph: 352-294-5971

Email: [djlemas@ufl.edu](mailto:djlemas@ufl.edu)

Abstract word count: 238

Abstract word limit: 250

Target Journal: GigaScience

Reference limit: not mention in the guideline

Reference count:

Total word limit: not mention in the guideline

Current word count:

**ABSTRACT**

***Background:***Untargeted metabolomics data processing remains a challenge due to the large size of metabolomics data files as well as the extensive steps and computational power required to process the data. Developed tools for metabolomics data processing are either local-based or web-based. Nevertheless, Local machines do not have enough computational power to process large dataset and web-based platforms needs tons of time to upload and process large data without closing the browser. Fortunately, cloud servers are available nowadays, whose configuration is good enough to process large data and all the jobs can be submitted to the server and kept running remotely even when our local machine is turned off. However, none of the untargeted metabolomics data processing tools was designed for cloud servers. Before using cloud servers, people need to specify a configuration, and better configuration costs more. Therefore, it is necessary to develop a tool specifically for cloud servers meanwhile able to guide the choice of configurations so that it will be easier for researchers to make budgets and process metabolomics data efficiently.

***Findings****:* Nextflow, a pipeline development tool supporting containerization, parallelization and high performance computing, was used to build the pipeline. Several tools and codes were connected and incorporated into Nextflow. Suggestions regarding the choice of cloud server configurations were made based on Nextflow report.

***Conclusions****:* An containerized untargeted metabolomics data processing tool specifically for cloud server was developed to facilitate metabolomics data analysis and collaborative research.

**1.1 Background**

**1.2 Findings**

1.2.1 The [Name] workflow

(A figure describing the workflow)

1.2.2 Implementation

1.2.3 [Name] Input/Output

1.2.4 Results

1.2.4.1 Simulation study

1.2.4.2 Real-world case study

1.2.5 Discussion

**1.3 Methods**

1.3.1 Data availability

1.3.1.1 Simulation study

1.3.1.2 Real-world case study

1.3.2 Data analysis

1.3.2.1 Simulation study

1.3.2.2 Real-world study

**1.7 Availability of source code**

**1.8 Availability of supporting data**

**1.9 Abbreviations**

**1.10 Competing interests**

The authors have no financial or personal relationships with other people or organizations that could inappropriately influence (bias) their work.

**1.11 Funding**

This work was supported by the National Institute of Diabetes and Digestive and Kidney Diseases [K01DK115632] and the University of Florida Clinical and Translational Science Institute [UL1TR001427]. The content is solely the responsibility of the authors and does not necessarily represent the official views the University of Florida’s Clinical and Translational Science Institute, or the National Institutes of Health.

**1.7 Author contributions**

**1.8 Acknowledgements**

**1.8 References**

**FIGURES**

Screenshot of log file

Relationship between input size and memory cost

Relationship between input number of samples and memory cost

**TABLES**

**SUPPLEMENTARY MATERIAL**