

Bringing Julia to Health: A Concise Overview of JuliaHealth

Jacob S. Zelko¹, Carlos Castillo-Passi², Jakub Mitura³, Divyansh Goyal⁴, Kosuri Lakshmi Indu⁵, and Jay Sanjay Landge⁶

¹CEPHIL, University of Saskatchewan

²Department of Radiology, Stanford University

³University Clinic for Radiology and Nuclear Medicine, Otto-von-Guericke-Universität

⁴Guru Gobind Singh Indraprastha University

⁵BVRIT Hyderabad College of Engineering for Women

⁶Indian Institute of Technology (IIT), Hyderabad

ABSTRACT

JuliaHealth is an open-science, volunteer-driven ecosystem within the Julia community that aims to bring high-performance, open-source computing to health research. This paper provides a concise overview of JuliaHealth, with a focus on its medical imaging and observational health ecosystems, and situates these efforts within the broader Julia ecosystem. We conclude by outlining future development goals and directions for community growth.

Keywords

health informatics, medical imaging, open source communities

1. Introduction

Founded in 2020, JuliaHealth was created to support the development and adoption of high-performance computing methods from the Julia ecosystem for biomedical and health research. The community's packages are developed under the JuliaHealth GitHub organization. Initially, JuliaHealth served as a coordination point for Julia community efforts related to the COVID-19 pandemic. Since then, JuliaHealth has continued to grow, bringing together dozens of active contributors across its GitHub repositories and a broader user community across Julia communication channels, and supporting research and professional activities across multiple venues [3, 4]. As the community has expanded, several sub-communities have organically emerged within JuliaHealth to address different aspects of health research, each with distinct computational needs and levels of maturity. This growth highlights both the opportunities and challenges of developing interoperable, reproducible software workflows in biomedical research. The purpose of this paper is to describe these emerging ecosystems, their current capabilities, their near- and long-term community goals, and how JuliaHealth integrates with the broader Julia ecosystem.

2. Statement of Need

Biomedical and health research increasingly relies on computational workflows that must be open, reproducible, and efficient. However, the software ecosystem supporting these workflows remains fragmented across closed-source platforms and multiple pro-

gramming languages. Researchers commonly combine tools written in MATLAB, Python, R, or C/C++, resulting in workflows that are difficult to integrate, reproduce, and maintain, particularly when performance-critical components are involved.

Julia provides a suitable technical basis for addressing these challenges because it supports high-level scientific programming with performance approaching low-level languages, enabling researchers to prototype and deploy within a single language. When external tools are needed, Julia can also reuse established biomedical software through interoperability packages such as Python-Call.jl and RCall.jl.

JuliaHealth addresses the need for a community-maintained, open-source framework that spans multiple packages and can be composed into end-to-end workflows (e.g., simulation to analysis, or cohort definition to study execution) while remaining transparent and reproducible. By coordinating development within a shared organization, JuliaHealth lowers the cost of discovering, combining, and maintaining interoperable tools for health research.

3. Ecosystems within JuliaHealth

JuliaHealth includes several growing ecosystems. Here, we highlight the medical imaging and observational health analysis ecosystems, which are currently among the more mature areas of JuliaHealth. We then briefly note other developing efforts. All packages mentioned are under active development, with varying levels of maturity.

3.1 Medical Imaging

The JuliaHealth medical imaging ecosystem supports end-to-end medical imaging research workflows with high-performance, open-source tools. The ecosystem spans physics simulations, data acquisition, and reconstruction to image processing, segmentation, and interpretation.

3.1.1 Image Acquisition and Simulation. The core package for medical imaging simulation in JuliaHealth is KomaMRI.jl [1], an open-source framework for general MRI simulation on CPUs and GPUs. It solves the Bloch equations with configurable numerical accuracy and provides a flexible environment for designing and optimizing MRI pulse sequences, including optional physiological motion models. KomaMRI.jl has already been used for car-

diac sequence optimization at low magnetic field strengths [2], synthetic dictionary generation for quantitative techniques, evaluation of hardware imperfections, and experimental imaging techniques for sensitively measuring oscillating magnetic currents in the brain. Its high-performance simulation core enables fast and reproducible exploration of sequence behavior, motion effects, and hardware limitations before scanner implementation, making it a powerful tool for physics-based optimization and machine learning applications.

3.1.2 Image Processing, Segmentation and Interpretation. For post-acquisition methods, JuliaHealth includes a comprehensive suite of packages for image I/O, visualization, segmentation evaluation, and workflow orchestration. For example, `MedPipe3D.jl` orchestrates `MedImages.jl`, `MedEye3D.jl`, and `MedEval3D.jl` into a workflow for data visualization, inspection, annotation, segmentation, and evaluation of 3D/4D images and other volumetric biomedical data (e.g., MRI, CT, PET). Beyond orchestration, I/O packages such as `DICOM.jl` support reading and writing Digital Imaging and Communications in Medicine (DICOM) images, metadata parsing, anonymization, and conversion to numerical arrays. Together, JuliaHealth offers a complete post-acquisition research environment that supports reproducible and scalable medical-imaging analysis and interpretation. Complementing imaging-focused workflows, JuliaHealth also supports population-level analyses through its observational health ecosystem.

3.2 Observational Health

The JuliaHealth observational health ecosystem supports patient population analysis and phenotype definition management. It also provides interoperability with Observational Health Data Sciences and Informatics (OHDSI) tools, handling of Observational Medical Outcomes Partnership Common Data Model (OMOP CDM; a standardized relational schema for observational health data) databases, and study creation and maintenance. Packages like `OMOPCDM-CohortCreator.jl`, `OMOPCommonDataModel.jl`, and `OHDSIAPI.jl` assist researchers in defining patient populations, managing OMOP CDM data, and integrating with external OHDSI tools. Furthermore, building on OMOP CDM databases, `OMOPCDMPathways.jl` maps and analyzes a patient's care continuum, while `OMOPCDMFeasibility.jl` characterizes patient cohorts based on phenotype definitions and statistical profiles across covariates of interest. Finally, `HealthBase.jl` provides templates for studies such as observational health studies and a common interface for packages to dispatch upon to bring unity to workflows across JuliaHealth. With these tools, JuliaHealth offers several options for conducting transparent, large-scale observational health analyses across institutions.

3.3 Other Developing Ecosystems

Beyond these two ecosystems, many parts of JuliaHealth are actively under development. For example, for simulation of biological and physiological processes, `Thunderbolt.jl` simulates cardiac electrophysiology and mechanics, and `BloodFlowTrixi.jl` can be used to explore hemodynamics. For interacting with third-party bioinformatics platforms, several API wrappers and specialized tools exist to bring relevant data or information from biomedical literature and chemical databases into one's workflow. These tools support programmatic retrieval and analysis via packages such as `PubMedMiner.jl` for accessing literature from PubMed/MEDLINE and `PubChemMiner.jl` for retrieving chemical structures and properties from PubChem. As JuliaHealth continues to grow, these ef-

forts in simulation modeling and integration with bioinformatics resources will help expand JuliaHealth into new ecosystems and further stages of development.

4. Discussion

While **JuliaHealth** is relatively young, it has a steady presence within the broader Julia community through its open-source package ecosystem, monthly community meetings, community programs such as Google Summer of Code (GSoC) and NumFOCUS-sponsored events, and discussion forums including Julia Discourse, Julia Slack, and Julia Zulip. Here, we highlight strategic partnerships JuliaHealth has been leveraging across the Julia ecosystem and community involvement.

4.1 Community Partnerships

Through cultivation of Julia community relationships, JuliaHealth has been able to expand its capabilities and support trans-organizational efforts in developing novel applications across disciplines. As a result, JuliaHealth has worked consistently with community organizations such as:

- BioJulia** - Fast, open, extensible software for bioinformatics and computational biology.
- JuliaEpi** - Open research community to develop an epidemiological modeling ecosystem written in Julia.
- JuliaImageRecon** - General purpose image reconstruction and computational imaging packages.
- MagneticParticleImaging** - Julia ecosystem for Magnetic Particle Imaging (MPI) simulation framework.
- MagneticResonanceImaging** - Collection of packages for MRI research and application (simulation, reconstruction and analysis).
- JuliaSurv** - Survival analysis tools and packages in Julia.

4.2 Community Outreach

JuliaHealth also seeks to foster a safe and welcoming environment through multiple communication platforms and community mentoring. We accomplish these goals by providing and promoting:

- monthly community meetings,
- community programs such as Google Summer of Code (GSoC) and NumFOCUS-sponsored events,
- discussion forums such as Julia Discourse, Julia Slack, and Julia Zulip,
- package development and planning on code hosting platforms such as the JuliaHealth GitHub, and
- the JuliaHealth blog to highlight ongoing work and community projects.

5. Future Community Development

As JuliaHealth continues to develop new tools and methodologies, some future areas of development are:

- combining census microdata alongside observational health data to explore social determinants of health,
- interactive community dashboards with custom JuliaHealth modules to host and present analysis results, and
- exploratory use of generative AI and large language models to support health research analysis and tool development.

6. Conclusion

As the ecosystem continues to grow, it will continue to provide a central presence within the Julia ecosystem to enable comprehensive research workflows and future software development. As active development continues, JuliaHealth offers a practical and extensible foundation for bringing high-performance and high-quality code to health research.

7. Acknowledgements

We warmly thank Dr. Dilum Aluthge for initiating JuliaHealth, the JuliaHealth and broader Julia community, and early contributions by Malina Hy, Varshini Chinta, and Fareeda Abdelazeez.

8. References

- [1] Carlos Castillo-Passi, Ronal Coronado, Gabriel Varela-Mattatall, Carlos Alberola-López, René Botnar, and Pablo Irarrazaval. KomaMRI.jl: An open-source framework for general MRI simulations with GPU acceleration. *Magnetic Resonance in Medicine*, 90(1):329–342, 2023. doi:10.1002/mrm.29635.
- [2] Carlos Castillo-Passi, Karl P. Kunze, Michael G. Crabb, Camila Munoz, Anastasia Fotaki, Radhouene Neji, Pablo Irarrazaval, Claudia Prieto, and René M. Botnar. Highly efficient image navigator based 3D whole-heart cardiac MRA at 0.55T. *Magnetic Resonance in Medicine*, 93(2):689–698, 2025. doi:10.1002/mrm.30316.
- [3] Jacob S. Zelko and Carlos Castillo Passi. Juliahealth mini-symposium. In *JuliaCon 2025*, 2025.
- [4] Jacob S. Zelko, Varshini Chinta, Malina Hy, and Fareeda Abdelazeez. The use of the julia programming language for global health informatics and observational health research. In *OHDSI Global Symposium*, 2023.