

Open and reproducible research in musculoskeletal imaging: why it matters and how to implement it with the guidelines of the Open and Reproducible Musculoskeletal Imaging

Original

Open and reproducible research in musculoskeletal imaging: why it matters and how to implement it with the guidelines of the Open and Reproducible Musculoskeletal Imaging Research (ORMIR) community / Bonaretti, S., Barzegari, M., Bevers, M., Boyd, S., Burghardt, A.J., Cameron, D., Chimento, F., Crimi, G., Degenhart, G., Durongbhan, P., Espinosa Hernandez, M.A., Fraterrigo, G., Ghasem-Zadeh, A., Grassi, L., Hirvasniemi, J., Hosseinitabatabaei, S., Iori, G., Kok, J., Kuczynski, M., Lee, Y., et al.. - In: JBMR PLUS. - ISSN 2473-4039. - 10:4(2026). [10.1093/jbmrpl/ziag025]

Availability:

This version is available at: 11583/3009779 since: 2026-04-10T10:51:48Z

Publisher:

Oxford University Press

Published

DOI:10.1093/jbmrpl/ziag025

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

Open and reproducible research in musculoskeletal imaging: why it matters and how to implement it with the guidelines of the Open and Reproducible Musculoskeletal Imaging Research (ORMIR) community

Serena Bonaretti^{1,*} , Mojtaba Barzegari² , Melissa Bevers^{3,4,5} , Steven Boyd^{6,7} , Andrew J. Burghardt⁸ , Donnie Cameron⁹ , Francesco Chiumento¹⁰ , Gianluigi Crimi¹¹, Gerald Degenhart¹² , Pholpat Dulongbhan¹³ , Michelle Alejandra Espinosa Hernandez^{13,14} , Giulia Fraterrigo¹¹ , Ali Ghasem-Zadeh¹⁵ , Lorenzo Grassi¹⁶ , Jukka Hirvasniemi¹⁷ , Seyedmahdi Hosseinatababaei¹⁸ , Gianluca Iori¹⁹ , Joeri Kok⁵ , Michael Kuczynski^{20,7} , YoungJun Lee²¹ , Cecilia Liberati²² , Sarah Manske^{6,7} , Matt McCormick²³ , Maria Monzon²⁴ , Martino Pani²⁵, Simone Poncioni²⁶ , Jilmen Quintiens²² , Sabine Rauber²⁷ , Paul Ritsche²⁸ , Alfonso Dario Santamaria¹¹, Francesco Santini²⁷ , Fabio Sarto²⁹ , Enrico Schileo¹¹, Vincent Stadelmann³⁰ , Kathryn S. Stok¹³ , Rachel Surowiec²¹ , Fulvia Taddei¹¹, Jared Vicory³¹, Matthias Walle⁷ , Mariska Wesseling³² , Danielle Whittier^{7,20}, Bettina Willie³³ , Andy Kin On Wong^{34,35} , Dženan Zukić³¹ 

¹Independent Researcher, Zurich, 8000, Switzerland

²Department of Chemical Engineering and Chemistry, Eindhoven University of Technology (TU/e), Eindhoven, 5612 AZ, The Netherlands

³Department of Internal Medicine, VieCuri Medical Center, Venlo, 5912 BL, The Netherlands

⁴NUTRIM Institute of Nutrition and Translational Research In Metabolism, Maastricht University, Maastricht, 6229 ER, The Netherlands

⁵Department of Biomedical Engineering, Eindhoven University of Technology, Eindhoven, 5612 AZ, The Netherlands

⁶Department of Radiology, Cumming School of Medicine, University of Calgary, Calgary, AB T2N 2T8, Canada

⁷McCaig Institute for Bone and Joint Health, University of Calgary, Calgary, AB T2N 4Z6, Canada

⁸Department of Radiology & Biomedical Imaging, University of California, San Francisco, San Francisco, CA 94143, United States

⁹Department of Medical Imaging, Radboud University Medical Center, Nijmegen, 6525 GA, The Netherlands

¹⁰School of Electronic Engineering, Dublin City University (DCU), Dublin, D09 V209, Ireland

¹¹Bioengineering and Computing Laboratory, IRCCS Istituto Ortopedico Rizzoli, Bologna, 40136, Italy

¹²Core Facility MicroCT, University Clinic for Radiology, Medical University Innsbruck, Innsbruck, 6020, Austria

¹³Department of Biomedical Engineering, The University of Melbourne, Carlton, VIC 3053, Australia

¹⁴Rehabilitation Sciences Institute, University of Toronto, Toronto, ON M5G 1V7, Canada

¹⁵Departments of Endocrinology and Medicine, Austin Health, The University of Melbourne, Heidelberg, VIC 3084, Australia

¹⁶Department of Biomedical Engineering, Lund University, Lund, 223 63, Sweden

¹⁷Department of Radiology & Nuclear Medicine, Erasmus MC University Medical Center Rotterdam, Rotterdam, 3015 GD, The Netherlands

¹⁸Department of Bioengineering, Faculty of Engineering, McGill University, Montreal, QC H3A 0E9, Canada

¹⁹Center for Photon Science, Paul Scherrer Institute PSI, Villigen, 5232, Switzerland

²⁰Department of Cell Biology and Anatomy, Cumming School of Medicine, University of Calgary, Calgary, AB T2N 2T8, Canada

²¹Weldon School of Biomedical Engineering, Purdue University, West Lafayette, IN 47907, United States

²²Department of Mechanical Engineering, KU Leuven, Leuven, 3001, Belgium

²³Fideus Labs, Durham, NC 27713, United States

²⁴Department of Health Science and technology, ETH Zurich, Zurich, 8092, Switzerland

²⁵School of Electrical and Mechanical Engineering, University of Portsmouth, Portsmouth, PO1 3DJ, United Kingdom

²⁶ARTORG Center for Biomedical Engineering Research, University of Bern, Bern, 3008, Switzerland

²⁷Department of Biomedical Engineering, University of Basel, Basel, 4123, Switzerland

²⁸Department of Sport, Exercise and Health, University of Basel, Basel, 4052, Switzerland

²⁹Department of Biomedical Sciences, University of Padova, Padova, 35131, Italy

³⁰Department of Research and Development, Schulthess Klinik, Zurich, 8008, Switzerland

³¹Medical Computing, Kitware, Inc., Carrboro, NC 27510, United States

³²Department of Orthopedics and Sports Medicine, Erasmus Medical Center, Rotterdam 3000 CA, The Netherlands

³³Faculty of Dental Medicine and Oral Health Sciences, McGill University and Shriners Hospital for Children-Canada, Montreal, QC H4A 0A9, Canada

Received: 1 October 2025. Revised: 5 February 2026. Accepted: 9 February 2026

© The Author(s) 2026. Published by Oxford University Press on behalf of the American Society for Bone and Mineral Research.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted reuse, distribution, and reproduction in any medium, provided the original work is properly cited.

³⁴Joint Department of Medical Imaging; Schroeder Arthritis Institute, University Health Network, Toronto, ON M5G 2C4, Canada

³⁵Dalla Lana School of Public Health, University of Toronto, Toronto, ON M5T 3M7, Canada

*Corresponding author: Serena Bonaretti, (serena.bonaretti.research@gmail.com).

Abstract

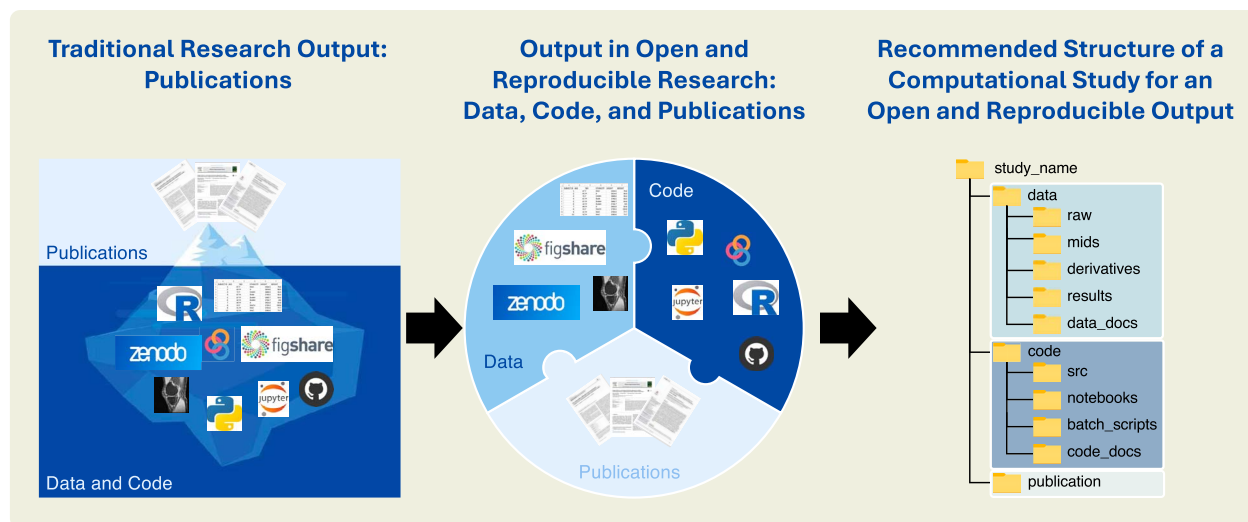
The Open and Reproducible Musculoskeletal Imaging Research community is a scientific community dedicated to promoting openness and reproducibility in musculoskeletal imaging, image processing, and computational modeling. In this perspective paper, we outline the motivations for conducting transparent research and provide practical guidelines for implementing it. We start by defining open and reproducible research and describing the benefits and challenges of working transparently. Next, we redefine the outputs of a computational research study as—ideally—a combination of data, code, and a publication, recommend a folder and file structure that reflects these three study outcomes, and describe how to maintain and update such a structure during the study and at study publication. Finally, we emphasize that working in an open and reproducible manner is a learning process, and the best way to acquire the necessary competencies is simply to start.

Keywords Open research, reproducible research, data sharing, open-source code, computational narratives, guidelines, license, ORMIR

Lay Summary

The Open and Reproducible Musculoskeletal Imaging Research community promotes openness and reproducibility in musculoskeletal imaging research. In this perspective paper, we explain why transparency matters and recommend how to conduct a computational study in an open and reproducible manner, focusing on its three outputs: data, code, and the publication. Finally, we highlight that the best way to learn these practices is simply to start.

Graphical Abstract



Introduction

In 2012, Panagiotopoulou *et al.*¹ published an article titled “What makes an accurate and reliable subject-specific finite element model? A case study of an elephant femur.” In this work, they sought “to construct the first reliable finite element model of the femur of an adult Asian elephant” from CT images. As part of the study, the authors conducted convergence analyses for the size, type, and number of elements used in the geometry discretization of the femur anatomy. In the following years, the paper received 33 citations (retrieved from Google Scholar: <https://scholar.google.com>. Accessed July 21, 2025). Some of these publications referred to the methodology and findings of this study in their introduction or discussion, while other publications built upon

the study methodology and findings. Among the latter, in 2018, Brassey *et al.*² published a study in which they used the same size, type, and number of elements as in Panagiotopoulou’s publication, motivating that “previous research *has found* such meshes perform well” and “strain values *have been found* to converge when element numbers exceed 200,000.” While building upon previous work is typical in research, in this case, there was an unexpected outcome. In 2014—four years before Brassey’s publication—Panagiotopoulou’s paper had been self-retracted because the authors “uncovered problems with the methods [...] and, therefore, some of the final data.”³

This story highlights three important considerations. First, Panagiotopoulou *et al.* deserve high praise for their scientific integrity. Second, what occurred to Brassey *et al.* could happen to

any researcher who builds a study on previous findings. Third, we researchers can be inspired to ask fundamental questions about some practical needs that we face in our daily work: How can we and our collaborators create and publish verifiable methodologies and findings that others can build upon? How can we avoid building our studies on incorrect methods and results? And how can we check the correctness of methodologies and results when we review a paper and assess its scientific value? The answer to these questions is: open and reproducible research.

Open and reproducible research: definitions, benefits, and challenges

Openness refers to “an approach to research based on open cooperative work that emphasizes the sharing of knowledge, results and tools as early and widely as possible”.⁴ Reproducibility—intended here as *computational* reproducibility—is the ability of researchers to duplicate the results of a previous study using the “same input data, computational steps, methods, and conditions of analysis”.⁵ Together, openness and reproducibility constitute a *way of working* whose ultimate goal is to improve the reliability and efficiency of scientific progress.^{6,7}

The scientific advantages of open and reproducible research are several and include evaluating the correctness of scientific claims,⁸ building on previous work with confidence and efficiency⁹—by reusing and extending existing computational pipelines—and comparing and validating new methods and algorithms against existing ones without having to reimplement them.¹⁰ These scientific benefits are accompanied by academic advantages. Publishing open data can increase the number of citations by approximately 70%, regardless of journal impact factor, author country of origin, and date of publication.¹¹ Similarly, making code available online can increase citations by a factor of three.¹² Furthermore, research laboratories can greatly benefit from adopting open and reproducible practices. Such a transparent methodological approach fosters an intrinsic “discipline” that supports the continuity of maintenance and extension of computational tools created in the labs, which would otherwise risk a short lifespan as they are usually developed by successive and distinct generations of PhD students and postdocs.⁶

Despite the aforementioned advantages, working in an open and reproducible manner has been challenging. Until a few years ago, guidelines were scarce, computational tools were not fully developed, and permanent, large, and free repositories for data and code were unavailable.¹³ However, nowadays there are numerous sources of information on open and reproducible research. The “Ten simple rules” series includes guidelines on good research practice,¹⁴ open development of scientific software,¹⁵ and effective research data management.¹⁶ Mature executable environments are available, such as Binder (<https://mybinder.org>), and established repositories, such as Zenodo (<https://zenodo.org>), facilitate the sharing of data and code. In addition to these practical challenges, many scientists were concerned about not receiving proper recognition, having to allocate additional time and effort to prepare data and code for release, and providing advantages to competing datasets or software when releasing seminal work.¹⁷ However, nowadays,

there are funding programs for open science, supported by both agencies (eg, Canadian Institutes of Health Research) and foundations (eg, Chan Zuckerberg Initiative), and reward practices involving hiring criteria and credit of scholarly outputs.¹⁸ Despite these considerable advances, the most critical limitation to sharing data and code may be related to the way we conceive the outcomes of computational studies.

What is the outcome of a computational research study?

Traditionally, the main outcome of a computational research study is a scientific paper. As we know, papers are concise reports limited by a defined number of words (typically 3000-6000) and a restricted number of figures and tables (usually 3-6). To meet these constraints, only a selection of the methods and results developed and obtained during a study are presented in a paper. Crucial implementation details (eg, computational initializations and parameters) are often omitted,¹² as well as intermediate or individual findings, because results are usually presented as aggregated statistics, such as means and standard deviations. In this context, a scientific paper does not represent “the scholarship itself, it’s merely scholarship advertisement”, as Donoho famously wrote in 2009.⁶ In other words, a scientific paper is just the tip of the iceberg representing research efforts.¹⁹ The actual work—that is, the data and code—remains hidden underwater (Figure 1, left).

In open and reproducible research, data and code emerge to be an integrative part of the outcome of a computational study. Together with publications, they constitute the three essential, interlocked pieces of the puzzle necessary to understand and reproduce the entire study (Figure 1, right). From the raw data of the input, through the code, researchers can recreate the results presented in the publication, and subsequently reuse and extend the workflow for new studies. However, for this to be feasible, data and code should be clearly structured and documented.²⁰

More recently, numerous guidelines have become available on how to publish computational studies in an open and reproducible manner while enabling attribution and facilitating sharing.^{8,9,15,21,22} Moreover, researchers in many fields have formed scientific communities to establish consensus-based standards and develop computational tools specific to their disciplines.²³⁻²⁵

ORMIR guidelines for open and reproducible computational studies in MSK imaging research

In musculoskeletal (MSK) imaging, the Open and Reproducible Musculoskeletal Imaging Research (ORMIR) community was founded in 2020, with the aims of creating open, reproducible, well-tested, and well-documented code to analyze MSK images; standardizing data acquisition and management to favor data sharing and algorithm comparison; and promoting a culture of openness and reproducibility in musculoskeletal imaging, image processing, and computational modelling for a faster advancement of the field.²⁶ The community is currently composed of approximately 70 international researchers from academia

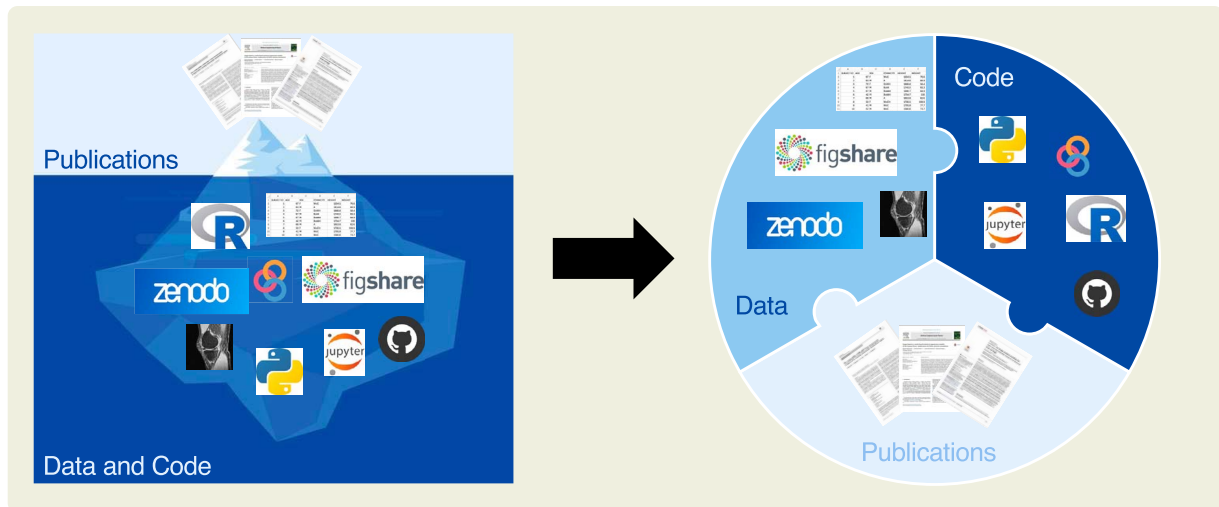


Figure 1 Representation of research outcomes. Traditionally, publications are the main research outcome. They constitute the shared, visible tip of an iceberg, whereas data and code remain hidden in the lab (left; modified from <https://stodden.net/AMP2011/slides/2011-amp-reproducible-research.pptx.pdf>). Conversely, in open and reproducible research data, code, and publication constitute the three interlocking pieces of the puzzle and are essential to each other to represent the entire study (right).

and industry at various career stages who have been active in creating guidelines, templates, and software to manage and analyze MSK images. At the moment, the following working groups are active: biomechanics, high-resolution CT, knee, muscle, data management and sharing, and spine—see www.ormir.org.

In the next two sections of this article, we will present the ORMIR guidelines on how to create an open and reproducible computational study using MSK images. These guidelines are based on the principle that open and reproducible research constitutes a methodological approach that should be implemented throughout an entire study, rather than being considered as a time- and resource-intensive task at publication. In addition, these guidelines are intended as a recommendation for future projects rather than strict rules, and they can be adapted to the specific needs of any computational study. ORMIR remains open to different folder structures that predate these guidelines or that are determined by project-specific requirements. In the following, first, we will focus on how to conduct a computational MSK research study for an open and reproducible publication. Then, we will provide suggestions on how to integrate and share the three study outcomes at publication. All recommendations are summarized in [Figure 2](#).

Conducting a computational MSK research study for an open and reproducible publication

Begin by organizing your research project into a folder structure similar to the one in [Figure 3](#). You can download the folder structure from Zenodo (www.doi.org/10.5281/zenodo.17206440) or implement it manually. Start with a main folder and name it after the study. Within the main folder, create three additional folders, each for one of the three study outcomes: data, code, and publication. The structure and content of each folder are described in more detail in the following sections.

Data

Within the data folder, add five folders: raw, mids, derivatives, and results for data at different stages of the reproducible workflow, and data_docs for documentation.

In the folder raw, store the original data of the study, that is, the image files—usually in DICOM or proprietary format—and demographic or clinical information of the subjects—commonly in tabular format. These data constitute the input of your workflow. Do not modify them either via computational or manual intervention to ensure the reproducibility of the whole study.^{8,27}

In the folder mids, restructure the files in raw following the specifications defined by the ORMIR Musculoskeletal Imaging Data Structure (ORMIR-MIDS),²⁸ which extends and adapts the Brain Imaging Data Structure (BIDS) principles²⁹ specifically for MSK imaging. The new structure comprises a hierarchy of nested folders arranged by subject, session, and imaging modality. It can be created using the ORMIR-MIDS Python package, which also converts the images from the Digital Imaging and Communications in Medicine (DICOM) format³⁰ to the Neuroimaging Informatics Technology Initiative (Nifti) format³¹ ([Figure 3](#), right). The Nifti format was chosen for compatibility with BIDS specifications and because it stores 3D/4D MSK imaging data into a single, standardized file, enabling efficient and reproducible processing.

During the study, in the folder derivatives, store the intermediate outputs of the computational workflow—such as segmentation masks, morphological measurements, and finite element analyses. Store also files that contain parameters used to modify the data contained in mids into the data contained in derivatives—eg, a .csv or .txt file containing the six parameters of a 3D image cropping. Saving these intermediate outputs will facilitate debugging and the potential reuse of partial findings.⁹

In the folder results, save the final output of the computational workflow, including tabular files reporting morphological, densitometric, mechanical, or statistical calculations across subjects. These are the data that you will present in the *Results* section of the publication.

| | Conducting a computational MSK research study for an open and reproducible publication | Publishing a computational MSK research study as open and reproducible |
|-------------|---|--|
| Data | <ul style="list-style-type: none"> Create 5 folders: raw, mids, derivatives, results, and data_docs <ul style="list-style-type: none"> Never modify the data in raw Use the ORMIR-MIDS structure in mids The data in results will appear in the publication Document the who/what/when/where/why/how Use open file formats and meaningful file names | <ul style="list-style-type: none"> Verify sharing feasibility: <ul style="list-style-type: none"> Consider consent and anonymization Consult the ORMIR Data Sharing Guidelines Select a repository providing DOI, version control, and embargo if needed Avoid links to cloud services or personal websites Choose an institutional or a Creative Commons license |
| Code | <ul style="list-style-type: none"> Create 4 folders: src, notebooks, batch_scripts, and code_docs <ul style="list-style-type: none"> Add the Python modules to src Use Jupyter Notebooks to analyze tabular data, generate graphs for publications, and create demos—use the ORMIR template Add the bash scripts to batch_scripts Document code using Sphynx, Read the Docs or Jupyter Book Write code for humans: use variables with descriptive names and follow style conventions Use version control (GitHub) | <ul style="list-style-type: none"> Change the GitHub repository visibility to public Use Binder to share notebooks interactively Host code documentation on GitHub or Read the Docs Sync the code with a repository providing DOI for citations Choose an open-source license |
| Publication | <ul style="list-style-type: none"> Write the manuscript in Latex, Markdown, or Google Docs | <ul style="list-style-type: none"> Add links to code and data repositories to the paper Add links to computational workflows to the graphs Upload to a preprint server Submit to an open access journal |
| Takeaways | <ul style="list-style-type: none"> It is a learning process, and you will learn by doing In case of doubts, ask the ORMIR community It does not have to be perfect. Just start! | <ul style="list-style-type: none"> Others can now build on your research You have contributed to a more transparent and faster advancement of MSK imaging research |

Figure 2 Summary of best practices for open and reproducible computational studies in musculoskeletal imaging research.

Finally, in the folder `data_docs`, add the data documentation, such as acquisition protocol, ethical approval, and any meta-data—that is, information about the data. As currently ORMIR does not provide specific guidelines for data documentation, simply collect all the reasonable information that specify the “who/what/when/where/why/how” of your dataset,¹³ so that other researchers can understand and reuse it. In general, provide information to make your data as FAIR (findable, accessible, interoperable, and reusable) as possible to maximize transparency, facilitate reuse, and support reproducibility.³²

At every stage of the computational workflow, use open and widely supported file formats.²⁷ For imaging data, prefer NIFTI or other open standards to scanner-specific proprietary formats. For tabular data, use comma-separated values (.csv) or tab-separated values (.tsv) over proprietary formats, such as Excel. Finally, give files meaningful names that clearly indicate their content.²⁷

Code

In the folder `code`, create the four folders `src`, `notebooks`, `batch_scripts`, and `code_docs`. In larger projects, add the additional folder `tests` for automated unit tests.³³

In the folder `src` (abbreviation for source), add the core algorithms of the study. Write code in the open-source programming language Python, which has gained popularity in medical imaging for the availability of general-purpose image processing packages—such as ITK³⁴ and SimpleITK³⁵—as well as deep learning and machine learning packages—including PyTorch,³⁶ TensorFlow,³⁷ and Scikit-learn.³⁸ Within `src`, organize your code into Python modules, that is, .py files. For larger projects, group the modules into a Python package and change the folder name to the package name, as required by the packaging tools.³⁹ If you are more familiar with other programming languages, including

open-source languages such as C++ or R, you may use them as well, provided that the code is well documented.

In the folder `notebooks`, add the study’s Jupyter Notebooks, which are the ideal format for integrating code with narrative, equations, and visualizations to effectively document reproducible workflows.⁹ Download the Jupyter Notebook template created by the ORMIR community from Zenodo (www.doi.org/10.5281/zenodo.17206440) and adapt it to your study. The template has a clear structure composed of aims, computational steps, outputs, and dependencies—that is, a print of the hardware characteristics and Python package versions.⁹ Use notebooks to analyze tabular data, generate figures for the publication, collect the main function calls from the modules in `src` into a computational workflow with narrative, and create examples or demos for your project.

In the folder `batch_scripts`, save the bash files. These files are the scripts that call the algorithms from the folder `src` and that you use for large-scale computations on servers.

In all cases, write code “for humans”, that is, make it readable rather than unnecessarily complex.⁴⁰ To create clear code, use variables with descriptive names that reflect their role in the computation⁴⁰ and adhere to the Python coding convention PEP8.⁴¹ Moreover, modularize code into functions—or classes—that represent units of operations.^{9,27} Document each function following the NumPy Style⁴² or Google Python Style⁴³—which clearly describe functionality, inputs, and outputs—and add inline comments to explain non-obvious steps and clarify the underlying logic. Finally, synchronize the code to a repository like GitHub (<https://github.com>), GitLab (<https://gitlab.com>), or similar. Such repositories provide version control to preserve the history of changes and revert to previous code in case of errors, and they are convenient for coordinating programming tasks in collaborative software development.²¹

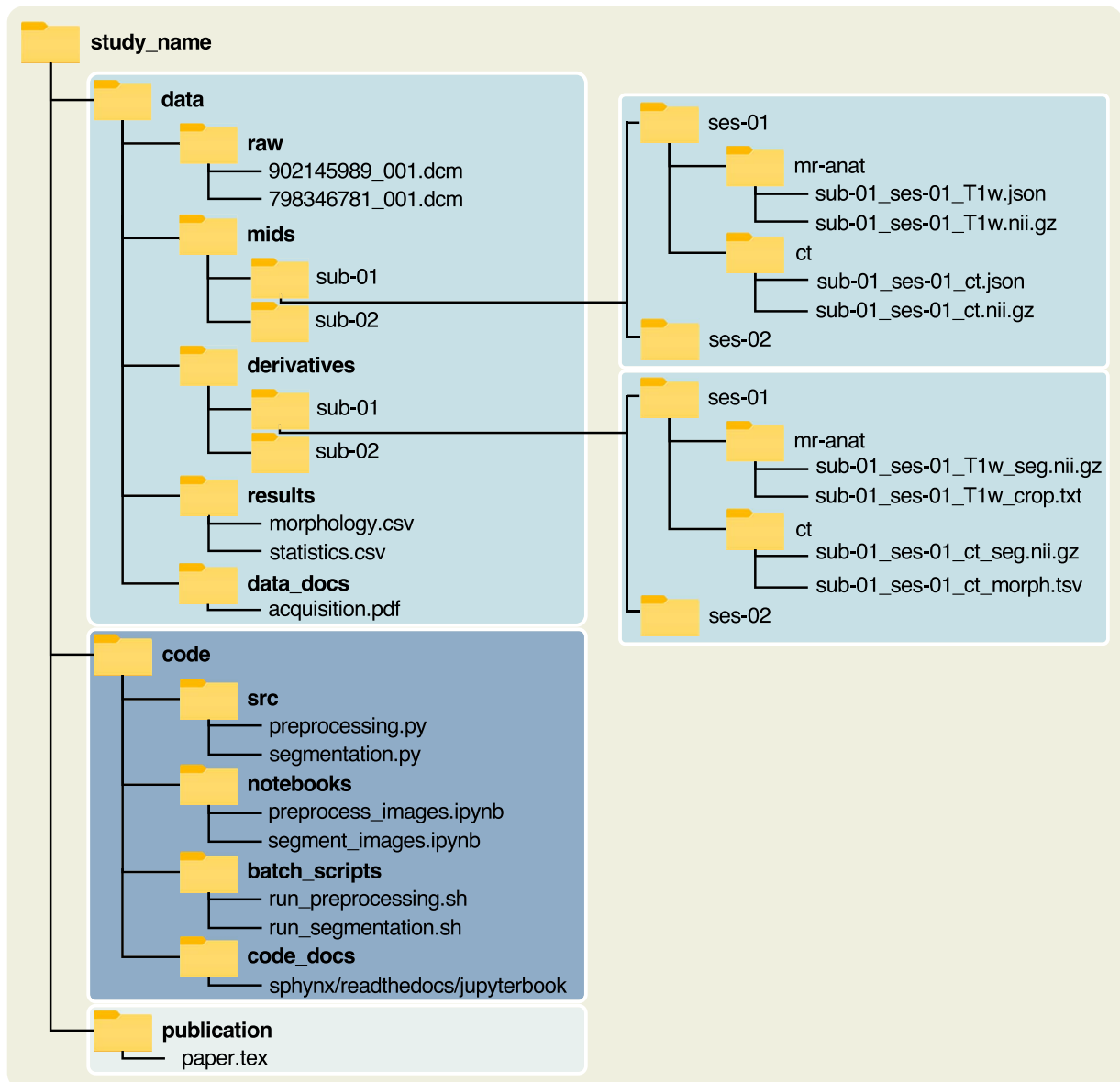


Figure 3 Folder and file structure of a computational research study for an open and reproducible publication. File extensions with acronyms expansions: `.dcm`: Digital Imaging and Communications in Medicine (standard file format for medical imaging data and associated metadata); `.json`: JavaScript Object Notation; `.nii.gz`: Neuroimaging Informatics Technology Initiative (compressed file format used to store medical imaging volumes); `.txt`: Plain Text; `.tsv`: Tab-Separated Values; `.csv`: Comma-Separated Values; `.pdf`: Portable Document Format; `.py`: Python Script; `.ipynb`: Interactive Python Notebook; `.sh`: Shell Script (script file containing commands for a Unix/Linux shell); and `.tex`: TeX (source file for documents written using the LaTeX typesetting system).

Finally, in the folder `code_docs` (or simply `docs`) create code documentation for developers and users, especially if you are building a large project.⁴⁴ In the documentation for developers, carefully explain the technical details of the code. Create this documentation using Sphinx⁴⁵ or Read the Docs⁴⁶ as they are compatible with the NumPy and Google Python styles. In the documentation for users, explain in nontechnical language how to install and execute your code and provide examples or demos. Write this documentation using Jupyter Book.⁴⁷

Publication

In the last folder `publication`, store all the material relative to the scientific paper. Write the manuscript in a format that permits

version control, such as LaTeX or Markdown, and opt for online tools that allow tracking changes and collaborative editing, such as Overleaf (www.overleaf.com), Google Docs (<https://docs.google.com>), Microsoft Word for the web (<https://word.cloud.microsoft>), or similar.²⁷

Publishing a computational MSK research study as open and reproducible

At publication, the primary task is to decide how and where to share the three study outcomes. Here are the details for each of them.

Data

Sharing study data requires attention to two main aspects: assessing the feasibility of sharing and choosing a data repository.^{48,49}

The feasibility of sharing depends on legal and ethical factors, including whether the study participants signed an appropriate consent form, the level of data anonymization or deidentification required, and the privacy legislation of your institution and country. These factors can preclude sharing some or all the content of the folders `raw`, `mids`, `derivatives`, and `data_docs`. However, you should be able to share the final aggregated data of your study—that is, the content of the folder `results`—which should not contain any personal data of the individual study participants. Further details are provided in the ORMIR Data Sharing Guidelines⁵⁰ (www.ormir.org/data_sharing_guidelines), which aim to clarify the main aspects involved in sharing MSK imaging data. These guidelines include information about informed consent, data ownership, and other legal and ethical considerations. For example, the consent form page provides templates of generic consent forms, including examples that are compliant and non-compliant with the European General Data Protection Regulation (GDPR), as well as institutional consent forms from the University of Calgary, Hamilton Health Sciences in Ontario, the UK Biobank, and the National Institutes of Health (NIH). The guidelines also address the issue of data ownership, which remains an unresolved problem. In some cases, data belong to the researchers and/or academic institutions supporting the research. In other cases, data belong to funding agencies sponsoring the projects (eg, the NIH in the United States). In still other cases, data are considered to belong to the study participants, who may have the right to request the erasure of their personal paper and electronic research records, as stipulated by the European GDPR. In general, before sharing data, discuss this possibility with your Principal Investigator, your institution, or your local Ethical Committee, who can provide appropriate legal and ethical advice.

To date, only a few dedicated MSK repositories exist where you can share your data, such as the Universal Musculoskeletal Ultrasound Repository (UMUD).⁵¹ If your data does not fit the scope of such repositories, share the data in your institution repository, following their regulation. Otherwise, use general purpose repositories such as Zenodo (<https://zenodo.org>)—where ORMIR is present as a community label—or Figshare (<https://figshare.com>). Both these repositories provide a digital object identifier (DOI) to ensure permanent availability and enable citation, versioning for data, and an embargo period if you cannot immediately share your data. Avoid links to personal websites or cloud storage, because these links tend to expire or be deleted.^{52,53}

Code

To make your code open, simply change the repository visibility from private to public on GitHub (or a similar platform). Complete the README.md file of the repository using the template provided by the ORMIR community (www.doi.org/10.5281/zenodo.17206440). Whenever possible, share the notebooks in an interactive and reproducible cloud environment, such as Binder (<https://mybinder.org>).⁵⁴ Host the code documentation created with Sphynx or Jupyter Book on GitHub Pages^{55,56} or the hosting service provided by Read the Docs. Finally, synchronize the GitHub repository with Zenodo,⁵⁷ Figshare,⁵⁸ or Software Heritage⁵⁹ to

obtain a DOI that enables reproducibility and citations. Note that sharing data and code alone may not be sufficient to guarantee inter-group computational reproducibility as software dependencies and execution environments may vary across institutions. Robust testing and documentation can significantly improve reproducibility in this respect, and guidelines such as those used by the Journal of Open Source Software (https://joss.readthedocs.io/en/latest/review_criteria.html) and the Journal of Open Research Software (<https://openresearchsoftware.metajnl.com/about/editorialpolicies>) can help support good testing practices.

Publication

Add the links to data and code repositories to the manuscript. If possible, add the specific links to the data and code used to create the graphs in their caption to enable readers to “recalculate the figure from all its data, parameters, and programs”.⁶⁰ Upload the paper to a preprint archive, such as arXiv (www.arxiv.org), bioRxiv (www.biorxiv.org), or medRxiv (www.medrxiv.org), and then submit it to a journal that supports open access. In the not too distant future, we may be able to write fully executable and interactive papers in which readers can rerun the described experiments.¹³ This vision is supported by existing pilot projects, where Jupyter Notebooks are submitted together with conference abstracts.^{61,62}

Licenses

Finally, add a license to your data, code, and publication to specify how you want them to be reused and modified. Without a license, you would retain all the rights and other researchers would not be allowed to use your material.²⁷ If available, use the licenses recommended by your institution. Otherwise, the ORMIR guidelines are reported below.

For data and publication, use the Creative Commons (CC) license, whose features—attribution (BY), Non Commercial use (NC), Share-Alike (SA), and No Derivative (ND)—can be combined to create a customized license. Creative Commons also provides the CC0 license, which permits releasing material to the public domain without retaining any copyright (<https://creativecommons.org/share-your-work/ccllicenses>).

For code, two main types of open-source licenses exist: restrictive—which extend openness to derived work (eg, GNU licenses)—and permissive, which do not require openness for derived work (eg, MIT, BSD, or Apache). For a complete list of licenses with their permissions and limitations, see <https://choosealicense.com/appendix>. Choose a license that is compatible with the licenses of the computational tools you used in your study and that you consider the most appropriate for the future of your project.²²

Just start!

Open and reproducible research is a way of working that supports our needs as researchers to assess scientific claims, build on previous reliable work, and contribute to faster advancement of our research field. In this perspective paper, we proposed concrete guidelines created by the ORMIR community on how to work in an open and reproducible manner. Adopting this methodological approach is a learning process that is most effective when learned

by doing. If you have doubts about some guidelines or tools, contact the ORMIR community. Do not aspire to perfection from the very beginning. Just start! Your efforts will prove valuable because other researchers will be able to deeply understand and build on your research, and you will have contributed to a more transparent and faster advancement of MSK imaging research.

Acknowledgments

We thank all the individuals who have contributed to and supported the open-source ecosystem, including developers, maintainers, and advocates, whose efforts have made open and reproducible research possible. Finally, we acknowledge the use of ChatGPT for editorial support, including grammar checking and flow enhancement.

Author contributions

Authorship was offered to any individual who has contributed to the development of the ORMIR community and its guidelines. Authors were asked to denote their contributions to the project according to the Contributor Roles Taxonomy (CRediT).

Serena Bonaretti (Conceptualization, Funding acquisition, Methodology, Project administration, Software, Supervision, Writing—original draft), Mojtaba Barzegari (Conceptualization, Software), Melissa Bevers (Writing—review & editing), Steven Boyd (Software, Writing—review & editing), Andrew J. Burghardt (Writing—review & editing), Donnie Cameron (Software, Writing—review & editing), Francesco Chimento (Writing—review & editing), Gianluigi Crimi (Software, Writing—review & editing), Gerald Degenhart (Writing—review & editing), Pholpat Durongbhan (Writing—review & editing), Michelle Alejandra Espinosa Hernandez (Writing—review & editing), Giulia Fraterrigo (Software, Validation, Writing—review & editing), Ali Ghasem-Zadeh (Writing—review & editing), Lorenzo Grassi (Writing—review & editing), Jukka Hirvasniemi (Software, Writing—review & editing), Seyedmahdi Hosseinatababaei (Writing—review & editing), Gianluca Iori (Data curation, Methodology, Software, Supervision), Joeri Kok (Writing—review & editing), Michael Kuczynski (Software, Writing—review & editing), YoungJun Lee (Writing—review & editing), Cecilia Liberati (Writing—review & editing), Sarah Manske (Conceptualization, Writing—review & editing), Matt McCormick (Conceptualization, Software, Writing—review & editing), Maria Monzon (Software, Writing—review & editing), Martino Pani (Writing—review & editing), Simone Poncioni (Software, Writing—review & editing), Jilmen Quintiens (Writing—review & editing), Sabine Räuber (Writing—review & editing), Paul Ritsche (Writing—review & editing), Alfonso Dario Santamaria (Software, Validation, Writing—review & editing), Francesco Santini (Conceptualization, Funding acquisition, Project administration, Software, Writing—review & editing), Fabio Sarto (Writing—review & editing), Enrico Schileo (Conceptualization, Supervision, Writing—review & editing), Vincent Stadelmann (Conceptualization, Funding acquisition, Writing—review & editing), Kathryn S. Stok (Conceptualization, Supervision, Writing—review & editing), Rachel Surowiec (Writing—review & editing), Fulvia Taddei (Conceptualization, Writing—review & editing), Jared Vicory (Software, Writing—review & editing), Matthias Walle (Writing—review & editing), Mariska

Wesseling (Software), Danielle Whittier (Supervision, Writing—review & editing), Bettina Willie (Writing—review & editing), Andy Kin On Wong (Supervision, Writing—review & editing), and Dženan Zukić (Supervision, Writing—review & editing)

Funding

The development of the ORMIR community and its guidelines has been supported by a NumFOCUS-funded Jupyter Community Workshop and a Swiss National Science Foundation grant (n. 221844).

Conflicts of interest

The authors have no competing interests to declare.

Data availability

No data were generated or analyzed in this paper.

References

1. Panagiotopoulou O, Wilshin SD, Rayfield EJ, Shefelbine SJ, Hutchinson JR. What makes an accurate and reliable subject-specific finite element model? A case study of an elephant femur. *J R Soc Interface*. 2012;9(67):351–361. <https://doi.org/10.1098/rsif.2011.0323>
2. Brassey CA, Gardiner JD, Kitchener AC. Testing hypotheses for the function of the carnivoran baculum using finite-element analysis. *Proc R Soc B Biol Sci*. 2018;285(1887):20181473. <https://doi.org/10.1098/rspb.2018.1473>
3. Panagiotopoulou O, Wilshin SD, Rayfield EJ, Shefelbine SJ, Hutchinson JR. What makes an accurate and reliable subject-specific finite element model? A case study of an elephant femur. *J R Soc Interface*. 2014;11(99):20140854. <https://doi.org/10.1098/rsif.2014.0854>
4. European Research Executive Agency. *Open Science*. Accessed August 27, 2025. https://rea.ec.europa.eu/open-science_en.
5. National Academies of Sciences, Medicine, Policy, Global Affairs, Board on Research Data, Division on Engineering, Physical Sciences, Committee on Applied, Theoretical Statistics, and Board on Mathematical Sciences. *Reproducibility and Replicability in Science*. National Academies Press; 2019. <https://doi.org/10.17226/25303>
6. Donoho DL, Maleki A, Rahman IU, Shahram M, Stodden V. Reproducible research in computational harmonic analysis. *Comput Sci Eng*. 2008;11(1):8–18. <https://doi.org/10.1109/MCSE.2009.15>
7. Munafò MR, Nosek BA, Bishop DVM, et al. A manifesto for reproducible science. *Nat Hum Behav*. 2017;1(1):0021. <https://doi.org/10.1038/s41562-016-0021>
8. Sandve GK, Nekrutenko A, Taylor J, Hovig E. Ten simple rules for reproducible computational research. *PLoS Comput Biol*. 2013;9(10):e1003285. <https://doi.org/10.1371/journal.pcbi.1003285>
9. Rule A, Birmingham A, Zuniga C, et al. Ten simple rules for writing and sharing computational analyses in jupyter notebooks. *PLoS Comput Biol*. 2019;15(7):e1007007. <https://doi.org/10.1371/journal.pcbi.1007007>

10. Freire J, Chirigati F. Provenance and the different flavors of computational reproducibility. *IEEE Data Eng Bull.* 2018;41(1):15 <http://sites.computer.org/debull/A18mar/p15.pdf>
11. Piwowar HA, Day RS, Fridsma DB. Sharing detailed research data is associated with increased citation rate. *PLoS One.* 2007;2(3):e308. <https://doi.org/10.1371/journal.pone.0000308>
12. Vandewalle P. Code sharing is associated with research impact in image processing. *Comput Sci Eng.* 2012;14(4):42–47. <https://doi.org/10.1109/MCSE.2012.63>
13. Gil Y, David CH, Demir I, et al. Toward the geoscience paper of the future: best practices for documenting and sharing research from data to software to provenance. *Earth Space Sci.* 2016;3(10):388–415. <https://doi.org/10.1002/2015EA000136>
14. Schwab S, Janiaud P, Dayan M, et al. Ten simple rules for good research practice. *PLoS Comput Biol.* 2022;18(6):e1010139. <https://doi.org/10.1371/journal.pcbi.1010139>
15. Prlić A, Procter JB. Ten simple rules for the open development of scientific software. *PLoS Comput Biol.* 2012;8(12):e1002802. <https://doi.org/10.1371/journal.pcbi.1002802>
16. Hassenstein MJ, Jung K. Ten simple rules for effective research data management. *PLoS Comput Biol.* 2025;21(12):e1013779. <https://doi.org/10.1371/journal.pcbi.1013779>
17. Stodden V. The scientific method in practice: reproducibility in the computational sciences. MIT Sloan Research Paper, (4773-10). 2010. <https://doi.org/10.2139/ssrn.1550193>
18. Kowalczyk OS, Lautarescu A, Blok E, Dall'Aglio L, Westwood SJ. What senior academics can do to support reproducible and open research: a short, 3-step guide. *BMC Res Notes.* 2022;15(1):116.
19. Freire J. A provenance-based infrastructure for creating reproducible papers. In: *Reproducible Research: Tools and Strategies for Scientific Computing. A Workshop in Association with Applied Mathematics Perspectives 2011.* University of British Columbia; 2011: <https://stodden.net/AMP2011/index.html>.
20. Chen X, Dallmeier-Tiessen S, Dasler R, et al. Open is not enough. *Nat Phys.* 2019;15(2):113–119. <https://doi.org/10.1038/s41567-018-0342-2>
21. Perez-Riverol Y, Gatto L, Wang R, et al. Ten simple rules for taking advantage of git and github. *PLoS Comput Biol.* 2016;12(7):e1004947. <https://doi.org/10.1371/journal.pcbi.1004947>
22. Morin A, Urban J, Sliz P. A quick guide to software licensing for the scientist-programmer. *PLoS Comput Biol.* 2012;8(7):e1002598–e1002597. <https://doi.org/10.1371/journal.pcbi.1002598>
23. Poldrack RA, Markiewicz CJ, Appelhoff S, et al. The past, present, and future of the brain imaging data structure (BIDS). *Imaging Neuroscience.* 2024;2:1–19. https://doi.org/10.1162/imag_a_00103
24. Gentleman RC, Carey VJ, Bates DM, et al. Bioconductor: open software development for computational biology and bioinformatics. *Genome Biol.* 2004;5(10):R80. <https://doi.org/10.1186/gb-2004-5-10-r80>
25. Odaka TE, Banihirwe A, Eynard-Bontemps G, et al. The Pangeo ecosystem: interactive computing tools for the geosciences: benchmarking on HPC. In: *Annual Workshop on HPC User Support Tools.*: Springer; 2019:190–204. https://doi.org/10.1007/978-3-030-44728-1_12.
26. Bonaretti S, Barzaghi L, Barzegari MM, et al. Introducing the open and reproducible musculoskeletal imaging research (ORMIR) community. 2023. <https://doi.org/10.5281/zenodo.8119243>
27. Wilson G, Bryan J, Cranston K, Kitzes J, Nederbragt L, Teal TK. Good enough practices in scientific computing. *PLoS Comput Biol.* 2017;13(6):e1005510. <https://doi.org/10.1371/journal.pcbi.1005510>
28. Bonaretti S, Espinosa Hernandez MA, Chiumento F, et al. ORMIR-MIDS: an open standard for curating and sharing musculoskeletal imaging data. In: *24th International Workshop on Quantitative Musculoskeletal Imaging (QMSKI).* The Barossa Valley; 2024: <https://doi.org/10.1093/jbmrpl/ziaf035>.
29. Gorgolewski KJ, Auer T, Calhoun VD, et al. The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments. *Sci Data.* 2016;3:1–9. <https://doi.org/10.1038/sdata.2016.44>
30. Bidgood WDJr, Horii SC. Introduction to the ACR-NEMA DICOM standard. *Radiographics.* 1992;12(2):345–355. <https://doi.org/10.1148/radiographics.12.2.1561424>
31. Cox RW, Ashburner J, Breman H et al. A (sort of) new image data format standard: NIFTI-1. In: *10th Annual Meeting of the Organization for Human Brain Mapping.* Vol. 22; 2004:01.
32. Wilkinson MD, Dumontier M, Aalbersberg IJJ, et al. The fair guiding principles for scientific data management and stewardship. *Sci Data.* 2016;3(1):1–9
33. Millman KJ, Pérez F. Developing open-source scientific practice. In: *Implementing Reproducible Research.* Chapman and Hall/CRC; 2018:149–183. <https://doi.org/10.1201/9781315373461-6>.
34. McCormick M, Liu X, Jomier J, Marion C, Ibanez L. Itk: enabling reproducible research and open science. *Front Neuroinform.* 2014;8:13. <https://doi.org/10.3389/fninf.2014.00013>
35. Lowekamp BC, Chen DT, Ibáñez L, Blezek D. The design of simpleitk. *Front Neuroinform.* 2013;7:45. <https://doi.org/10.3389/fninf.2013.00045>
36. Paszke A, Gross S, Massa F, et al. Pytorch: an imperative style, high-performance deep learning library. *Adv Neural Inf Proces Syst.* 2019;32. <https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html>
37. Abadi M, Agarwal A, Barham P, et al. Tensorflow: large-scale machine learning on heterogeneous systems. In: *12th USENIX Symposium on Operating Systems Design and Implementation.* 2015. <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf>
38. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: machine learning in python. *J Mach Learn Res.* 2011;12:2825–2830.
39. Python Packaging Authority. *Python Packaging User Guide.* Accessed August 28, 2025. <https://packaging.python.org/en/latest/>
40. Knuth DE. Literate programming. *Comput J.* 1984;27(2):97–111. <https://doi.org/10.1093/comjnl/27.2.97>
41. van Rossum G, Warsaw B, Coghlan A. *Pep8 - Style Guide for Python Code.* Accessed August 28, 2025. <https://peps.python.org/pep-0008/>.
42. *Numpy Style Guide.* Accessed August 28, 2025. <https://numpydoc.readthedocs.io/en/latest/format.html>

43. *Google Python Style Guide*. Accessed August 28, 2025. <https://google.github.io/styleguide/pyguide.html>
44. FdV Leprevost VC, Barbosa ELF, Perez-Riverol Y, Carvalho PC. On best practices in the development of bioinformatics software. *Front Genet*. 2014;5:199. <https://doi.org/10.3389/fgene.2014.00199>
45. Turner A, Tran B, Sewell C, et al. *Sphinx*. Accessed August 28, 2025. <https://www.sphinx-doc.org/en/master/index.html>.
46. *Read the Docs*. Accessed August 28, 2025. <https://about.readthedocs.com/>
47. Executable Books Community. *Jupyter Book*. Accessed August 28, 2025. <https://jupyterbook.org/en/stable/intro.html>.
48. Scheibner J, Ienca M, Kechagia S, et al. Data protection and ethics requirements for multisite research with health data: a comparative examination of legislative governance frameworks and the role of data protection technologies. *J Law Biosci*. 2020;7(1):05. <https://doi.org/10.1093/jlb/l5aa010>
49. Battle JC, Dreyer K, Allen B, et al. Data sharing of imaging in an evolving health care world: report of the ACR data sharing workgroup, part 1: data ethics of privacy, consent, and anonymization. *J Am Coll Radiol*. 2021; 18(12): 1646–1654. <https://doi.org/10.1016/j.jacr.2021.07.014>
50. Burghardt A, Wong AKO, Taddei F, et al. ORMIR data sharing guidelines - v1.1 (v1.1). *Zenodo*. 2025. <https://doi.org/10.5281/zenodo.17206129>
51. Ritsche P, Sarto F, Santini F, et al. UMUD: a web application for easy access to musculoskeletal ultrasonography datasets. 2025. Preprint. <https://doi.org/10.31219/osf.io/syr4z>
52. Dellavalle RP, Hester EJ, Heilig LF, et al. Going, going, gone: lost internet references. *Science*. 2003;302(5646):787–788. <https://doi.org/10.1126/science.1088234>
53. Klein M, Van de Sompel H, Sanderson R, et al. Scholarly context not found: one in five articles suffers from reference rot. *PLoS One*. 2014;9(12):e115253. <https://doi.org/10.1371/journal.pone.0115253>
54. Jupyter Project, Bussonnier M, Forde J, et al. Binder 2.0—reproducible, interactive, sharable environments for science at scale. In: *Proceedings of the 17th python in science conference, Scipy*. 2018; <https://proceedings.scipy.org/articles/Majora-4af1f417-011.pdf>
55. *GitHub Pages Documentation*. Accessed August 28, 2025. <https://docs.github.com/en/pages>.
56. *Jupyter Book - Publish your Book Online*. Accessed August 28, 2025. <https://jupyterbook.org/en/stable/start/publish.html>
57. *GitHub Docs - Referencing and Citing Content*. Accessed August 28, 2025. <https://docs.github.com/en/repositories/archiving-a-github-repository/referencing-and-citing-content>
58. *How to Connect Figshare with your GitHub Account*. Accessed August 28, 2025. <https://info.figshare.com/user-guide/how-to-connect-figshare-with-your-github-account/>
59. *Welcome to Software Heritage Documentation - Get a Permanent Identifier for your Source Code*. Accessed August 28, 2025. <https://docs.softwareheritage.org/#landing-basics>.
60. Claerbout JF, Karrenbach M. Electronic documents give reproducible research a new meaning. In: *SEG Technical Program Expanded Abstracts 1992*. Society of Exploration Geophysicists; 1992:601–604. <https://doi.org/10.1190/1.1822162>.
61. Caprarelli G, Sedora B, Ricci M, Stall S, Giampoala M. Notebooks now! The future of reproducible research. *Earth and space. Science*. 2023;10(12). <https://doi.org/10.1029/2023EA003458>
62. Erdmann C, Stall S, Gentemann C, et al. *Guidance for Agü Authors - Jupyter Notebooks*. 2021. <https://doi.org/10.5281/zenodo.5651648>