

Reproducibility in Biomedical Research: A Systems Prescription



Gray Babbs, PhD, MPH¹ , Ishani Ganguli, MD, MPH², and Alyssa Bilinski, PhD^{1,3}

¹Department of Health Services, Policy, and Practice, Brown University School of Public Health, Providence, RI, USA; ²Division of General Internal Medicine and Primary Care, Brigham and Women's Hospital, Harvard Medical School, Boston, MA, USA; ³Department of Biostatistics, Brown University School of Public Health, Providence, RI, USA

J Gen Intern Med
DOI: 10.1007/s11606-026-10536-x
© The Author(s) 2026

Rich public datasets and modern computing have transformed biomedical research. Scientists can now access much of the world's information in a few clicks and conduct complex analyses in seconds. However, these advances have also introduced new researcher degrees of freedom and opportunities for error, fueling growing concerns about reproducibility—the ability of new researchers to obtain the same result using the same data and methods as the original researchers.¹ Even though these critiques are themselves a product of science's unprecedented openness, they highlight the need for more robust quality assurance in an increasingly complex research landscape.

Where are we now? To illustrate current practices, we examined code-sharing and data-sharing practices from 3 months of Original Investigations and Research Letters ($N=134$) in *JAMA*, *JAMA Internal Medicine*, and *JAMA Pediatrics* (Online Supplement). Only 4% of publications shared code, and 5% shared data in public repositories. Code was listed as available upon request for another 10% of manuscripts and unavailable for 86%. Likewise, 16% reported public data but did not provide it in a replication package, 39% denoted data available upon request, and 40% listed data unavailable.

Why do 96% of papers fail to share code, and 95% fail to share data? These pervasive omissions point to structural failures rather than isolated individual choices. Reproducibility is impeded by gaps in training, limitations in infrastructure, and misaligned incentives. We describe these barriers and propose actions for researchers, journal editors, and funders.

BARRIERS TO REPRODUCIBILITY

The most fundamental barrier is training. Biomedical research increasingly requires code, but biomedical curricula rarely teach best coding practices. (None of the authors, for

instance, were taught to test code as part of coursework or training.) Writing code that performs as expected is notoriously difficult, even for trained software engineers, let alone researchers learning on the fly. In practice, coding is commonly delegated to a single researcher or small group, often students or junior staff, with limited oversight. Collaborators may review output, but formal code review or testing is rare.

Second, reproducibility standards in medical academia remain unclear. Even when teams implement internal review processes, these are rarely communicated in manuscripts, in contrast to careful documentation of data sources and statistical methods. Standards are omitted from checklists (e.g., CONSORT, CHEERS) that journals use to guide methodological reporting.

Third, preparing data and code for public dissemination takes time and effort that researchers have little incentive to spend. Under pressure to publish, publicly cataloging data and code is easy to deprioritize, especially when peers skip it. The NIH now requires Data Management and Sharing Plans for competing applications that generate scientific data. However, the NIH does not oversee implementation, and requirements do not cover studies using secondary data sources.²

Concerns about misuse and scrutiny compound this disincentive. Authors may fear that code could be misinterpreted or applied for unintended purposes. For example, authors may worry algorithms to identify transgender beneficiaries for research could be used to discriminate against trans patients.³ With the rise of large language models, authors may worry about limited control over intellectual property. Authors may fear that sharing data and code will expose them to additional criticism compared to peers who choose not to make resources public.

POTENTIAL STEPS

Researchers

Three practices borrowed from software engineering could substantially reduce coding errors in research: code testing, code review, and repeat coding. All require training and practice, and we emphasize the need to incorporate them into

formal biomedical research curricula and develop research-specific reporting standards.

Biomedical researchers typically rely on two implicit checks: that code runs without errors and that output aligns with expectations. However, these are both ad hoc and prone to confirmation bias, as researchers are more likely to scrutinize code that generates surprising results. In code testing, software engineers take a more systematic approach, defining the expected behavior of each function and writing tests to verify that given inputs produce correct outputs.⁴ For an example in a research context, when collapsing person-year-level data to the person level, a test could confirm that the number of unique individuals is preserved. Designing tests that anticipate what could go wrong is a skill that requires practice, but when well-implemented, can identify errors that may be otherwise difficult to catch.

Code review, a careful examination of data cleaning and analysis code by someone not involved in writing it, is a second safeguard. Though standard in software engineering, it remains rare in academic research.⁵ Its value lies not just in catching errors but also in the discipline it imposes: code written with the expectation that a colleague will read it tends to be cleaner and better documented. While industry best practices involve frequent review of small units of code, reviewing a complete analytic pipeline at a project's end may be preferable in an academic setting.⁵ As a side benefit, code review helps prepare code for public dissemination.

Last, repeat coding, multiple researchers independently coding the same analysis, is the most resource-intensive approach but also the most rigorous. When two researchers write code to answer the same question and arrive at different answers, the discrepancy may reflect a coding error. However, it may also reveal differing analytic assumptions: how to handle missing values or which observations to exclude. Either way, the disagreement is informative, surfacing bugs and hidden judgment calls.

The growing capacity of large language models (LLMs) to explain, critique, and write code has significant potential to lower the burden of all three practices. In addition to lowering the burden of initial code development (and proprietary sentiment toward code), LLMs can help generate tests, flag common errors, serve as a first-pass code reviewer, or aid in repeat coding. Creative application of these tools can make the gap between current and best practice less daunting to close.

Journal Editors and Funders

Individual practice only goes so far without institutional reinforcement. As de facto standard-setters for biomedical research, journal editors are well-positioned to raise expectations for code and data-sharing. While enforcing code-sharing (e.g., at the revised manuscript stage) is a start, some

journals go further. The *American Economic Review* requires authors to submit replication packages with code and data. When data are non-public, authors must indicate whether a private version can be made available to the Data Editor or a designated third-party replicator.⁶ *Bioinformatics* requires peer review of new software, algorithms, and code.⁷ To ease the burden on human reviewers, journals could also consider AI-powered review tools.

Last, funders, too, have a role: providing resources for replication packages, supporting double coding for high-stakes analyses, and coordinating shared infrastructure where useful. For example, funders could support the development of infrastructure for replicating studies that rely on restricted data sources such as Medicare claims, perhaps by employing HIPAA-compliant LLM coding support tools.

CONCLUSION

For individuals and research teams, the barriers to reproducibility can be substantial, but are not insurmountable. The practices outlined here—testing, reviewing, and sharing code—have been proven in other fields and can increasingly be supported by AI tools that lower the cost of adoption. Standardizing these practices is not just about rigor; it is a concrete mechanism for reinforcing the credibility and public trustworthiness of scientific results.

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s11606-026-10536-x>.

Corresponding Author: Gray Babbs, PhD, MPH; Department of Health Services, Policy, and Practice, Brown University School of Public Health, Providence, RI, USA (e-mail: gray_babbs@brown.edu).

Author Contribution Dr. Babbs, Dr. Ganguli, and Dr. Bilinski contributed to the conception, drafting, and revising of the work and provided final approval for publication.

Funding Dr. Bilinski was funded by the National Institute of General Medical Sciences (5R35GM155224).

Data Availability The data cited in this article are included in the supplement.

Declarations

Conflict of interest The authors declare that they do not have a conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material

is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

REFERENCES

1. Committee on Reproducibility and Replicability in Science, Board on Behavioral, Cognitive, and Sensory Sciences, Committee on National Statistics, et al. Reproducibility and Replicability in Science. National Academies Press; 2019:25303. <https://doi.org/10.17226/25303>.
2. National Institutes of Health. Writing a Data Management & Sharing Plan. NIH Grants and Funding. August 5, 2025. https://nihodoerco-mm.az1.qualtrics.com/jfe/form/SV_eypqaXlx2j1Y9T?Q_CHL=si&Q_CanScreenCapture=1. Accessed 8 Sept 2025.
3. **Alpert AB, Babbs G, Sanaeikia R, et al.** Doing Justice: Ethical Considerations Identifying and Researching Transgender and Gender Diverse People in Insurance Claims Data. *J Med Syst.* 2024;48(1):96. <https://doi.org/10.1007/s10916-024-02111-w>.
4. **Lucas TCD, Pollington TM, Davis EL, Hollingsworth TD.** Responsible modelling: Unit testing for infectious disease epidemiology. *Epidemics.* 2020;33:100425. <https://doi.org/10.1016/j.epidem.2020.100425>.
5. **Vable AM, Diehl SF, Glymour MM.** Code Review as a Simple Trick to Enhance Reproducibility, Accelerate Learning, and Improve the Quality of Your Team's Research. *Am J Epidemiol.* 2021;190(10):2172-2177. <https://doi.org/10.1093/aje/kwab092>.
6. American Economic Association. Data and Code Availability Policy. American Economic Association. 2026. <https://www.aeaweb.org/journals/data/data-code-policy>. Accessed 5 Feb 2026.
7. Oxford University Press. Instructions for Authors. *Bioinformatics.* 2025. https://academic.oup.com/bioinformatics/pages/instructions_for_authors. Accessed 31 Mar 2025.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.