of resilience. It is not. Precariousness describes how far a system state is from a tipping point, past which it could not recover. This could be measured as a simple Euclidean distance (in our simple bivariate space, as the distance from disturbed state to tipping point), or could be measured in terms of the magnitude of extra disturbance required to push the disturbed state past the tipping point (this requires reference to the relationship between system state and disturbance magnitude).

Furthermore, we wonder whether precariousness actually is, and should be, the most important component of resilience? In our own review of recent literature on ecological resilience we found very few examples of the measurement of precariousness or latitude, perhaps because tipping points are hard to measure. Tipping points and precariousness imply a rather binary, good/bad view of disturbed biological systems, which makes sense when systems are poorly understood, but should become much more nuanced when systems are measured and modelled well [5]. Y&R themselves note that 'critical slowing down' can happen, suggesting that behaviour near tipping points is linked not only to resistance but also to rates (and probabilities) of recovery. Critical speeding up also seems possible for very repellent tipping points, even if they can be crossed easily by large disturbances.

## Non-Equilibrium Dynamics Are Complicated

Y&R point out correctly that insults to biological systems can occur as sudden pulse events (e.g., hurricanes), as ramped change (e.g., invasion by exotic species; climate change), or as chronic change (e.g., constant harvesting regime), and that systems will respond differently to different types of insult. This raises another important semantic issue. In our own work on demography we distinguish between demographic disturbances (sudden events, followed by normal processes) and demographic perturbations (chronic changes to demographic processes) [6]. This follows the standard definition of disturbance as an event (usually singular) that causes a temporary period of environmental stress that upsets the normal state that a system is in. If the exogenous event is chronic, then the system processes have changed permanently, and the previously 'normal' state can only be returned to via a process of adaptation. We suggest that such chronic insults should be called perturbations or, simply, change. In a scenario of change, different measures of system response are required, and might be called 'robustness' and 'adaptation' rather than 'resistance' and 'recovery'.

Y&R also point out that disturbances have histories, and that states are rarely at equilibrium. This echoes our point that studies of resilience should recognise the amplitude, structure, and frequency of disturbances. The idea that impacts of disturbance might depend on initial system state is known as hysteresis [7]. Hysteresis is well known in ecological systems, particularly coral reefs [8], and is captured by the nonlinear resistance and recovery curves in our bivariate representation of resilience. Y&R suggest that disturbance itself is enough to completely change the resistance landscape of a biological system. This implies a very interesting feedback loop between disturbance and resilience, mediated by the adaptive dynamics of the system [9], which might complicate, but not deny, our bivariate approach.

## Studying Resilience Case-by-Case?

Y&R end their response by recommending that various measures of resilience can and should be tailored to the individual systems and questions being studied, while 'future' efforts should aim to find the smaller set of metrics required for resilience management. The problem we have with this approach is that the study of resilience in ecology and evolution already suffers from confusing terms, metrics, and definitions. Our goal was to encourage a standard approach such that understanding could be gained not only from individual systems but also from comparisons of resilience among systems. The study of resilience is no longer in its infancy, and there seems no better time to agree on a standard toolbox for measuring and modelling it. The bivariate approach that we suggested might not be 'it', but we would like to see the (currently precarious) study of resilience traverse its own tipping point towards a stable attractor of standardised empirical and theoretical research.

[1]Centre for Ecology and Conservation, University of Exeter, Penryn Campus, Penryn, Cornwall, TR10 9FE, UK

*Correspondence: d.j.hodgson@exeter.ac.uk (D. Hodgson).
http://dx.doi.org/10.1016/j.tree.2015.11.003

### References

1. Hodgson, D. *et al.* (2015) What do you mean, 'resilient'? *Trends Ecol. Evol.* 30, 503–506
2. Yeung, A.C.Y. and Richardson, J.S. (2016) Some conceptual and operational considerations when measuring 'resilience': a response to Hodgson *et al*. *Trends Ecol. Evol.* 31, 2–3
3. Gunderson, L.H. (2000) Ecological resilience – in theory and application. *Annu. Rev. Ecol. Syst.* 43, 425–439
4. Holling, C.S. (1996) Engineering resilience versus ecological resilience. In *Engineering within Ecological Constraints* (Schulze, P.C., ed.), pp. 31–44, National Academies Press
5. Capon, S.J. *et al.* (2015) Regime shifts, thresholds and multiple stable states in freshwater ecosystems; a critical appraisal of the evidence. *Sci. Total Environ.* 534, 122–130
6. Stott, I. *et al.* (2012) Beyond sensitivity: nonlinear perturbation analysis of transient dynamics. *Methods Ecol. Evol.* 3, 673–684
7. Beisner, B.E. *et al.* (2003) Alternative stable states in ecology. *Front. Ecol. Environ.* 1, 376–382
8. Mumby, P.J. *et al.* (2007) Thresholds and the resilience of Caribbean coral reefs. *Nature* 450, 98–101
9. Cole, L.E. *et al.* (2014) Recovery and resilience of tropical forests after disturbance. *Nat. Commun.* 5, 3906

---

Scientific Life

# Elevating The Status of Code in Ecology

K.A.S. Mislan,[1,2,*] Jeffrey M. Heer,[3] and Ethan P. White[4,5]

**Code is increasingly central to ecological research but often remains unpublished and insufficiently recognized. Making code available**

**allows analyses to be more easily reproduced and can facilitate research by other scientists. We evaluate journal handling of code, discuss barriers to its publication, and suggest approaches for promoting and archiving code.**

## The Role of Code in Ecological Research

Most ecologists now commonly write code as part of their laboratory, field, or modeling research. The transition to a greater reliance on code has been driven by increases in the quantity and types of data used in ecological studies, alongside improvements in computing power and software [1]. Code is written in programming languages such as R and Python, and is used by ecologists for a wide variety of tasks including manipulating, analyzing, and graphing data. A benefit of this transition to code-based analyses is that code provides a precise record of what has been done, making it easy to reproduce, adapt, and expand existing analyses.

Scientific code can be separated into two general categories – analysis code and scientific software. Analysis code is code that is used to correct errors in data, simulate model results, conduct statistical analyses, and create figures [2]. Release of analysis code is necessary for the results of a study to be reproducible [2]. The majority of code written for ecological studies is analysis code, and making this code available is valuable even if it is rough because it documents precisely what analyses have been conducted [2–4]. Scientific software is more general and is designed to be used in many different projects (e.g., R and Python packages). The development of ecological software is becoming more common and software is increasingly recognized as a research product [5,6].

## Current Standards for Code in Ecology

Journals are the primary method that ecologists use to communicate results of studies. Therefore, the way journals handle code is important for evaluating the current status of code in ecology. To explore the current status of code in ecology journals, we identified journals through a search of the Journal Citation Reports (JCR) using the following search terms: 'Ecology' for category, '2013' for year, 'SCIE' (Science Citation Index) and 'SSCI' (Social Sciences Citation Index) for editions checked, and 'Web of Science' for the category schema. We selected the top 100 results for analysis and, after excluding museum bulletins, a book, and a journal with broken website links, evaluated a total of 96 journals. We searched the author guidelines for each journal to determine if there was any mention of code or software in the context of scientific research. We also conducted more specific searches to determine if journals had a section for documentation of scientific software releases, and if journals had a policy requiring the release of code and/or data for article publication. Data release policies provide a useful comparison to code release policies because there have been ongoing efforts to encourage or require the release of data once results are published (e.g., [7]).

As of June 1, 2015, more than 75% of ecology journals do not mention scientific code in the author guidelines (Figure 1). Of the journals that mention scientific code, only 14% require code to be made available. Nearly threefold more journals (38%) require data to be made available. A very small subset of journals (7%) have created a special section for software releases or have added software releases to a list of options for existing methods sections (Figure 1). These findings are similar to a recent analysis of journal code policies in other scientific fields [8].

## Barriers to Publishing Code in Ecology

Elevating the status of code in ecology will require changes in attitude and policy by both journals and researchers. Researchers are often concerned about making their code public for a variety of reasons [4,9]. One of the main concerns is that publishing code takes time and researchers do not receive sufficient credit to justify this effort. This is compounded by concerns that releasing code may increase the risk of being scooped or hinder the researcher's (or their institution's) ability to commercialize the software [9]. In ecology, we believe that the benefits of publishing code outweigh the potential risks. There is little potential for commercialization of ecological analysis code, or even software, and reuse of code by others will raise the impact of the publications by the author of the code. It is also common for

### Trends

Code is frequently written for ecological studies.

Most ecology journals do not address code or software.

Journals can promote release of code by changing article formats and requirements.

Code archives should provide a license and be long-term and citable.

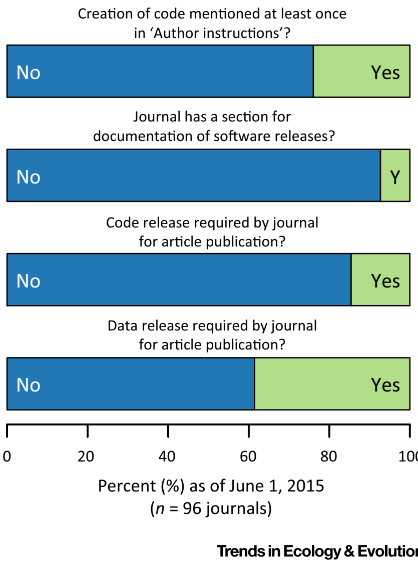


Figure 1. Current Status of Code in Ecology Journals. Most ecology journals do not have requirements or guidelines (as of June 1, 2015) for making code and data available. Ecology journals listed in the Journal Citation Reports (JCR) in 2013 were evaluated. Data and code available at http://dx.doi.org/10.5281/zenodo.34689.

scientists to believe that their code is not useful and that the description of what their code does (typically in the methods section of a journal article) is sufficient to allow the analysis to be reproduced. However, computational and statistical methods have become increasingly complicated, and access to the analysis code is now crucial to understanding precisely how analyses were conducted [2,4,9]. Even code that is rough and difficult to run on other systems (owing to software dependencies and differences in computing platforms) still provides valuable information as part of detailed documentation of the analyses [2,4,9]. Given the relatively low risk and potentially large benefit to science of releasing code, sufficient incentives are needed to motivate scientists to take the time to do so.

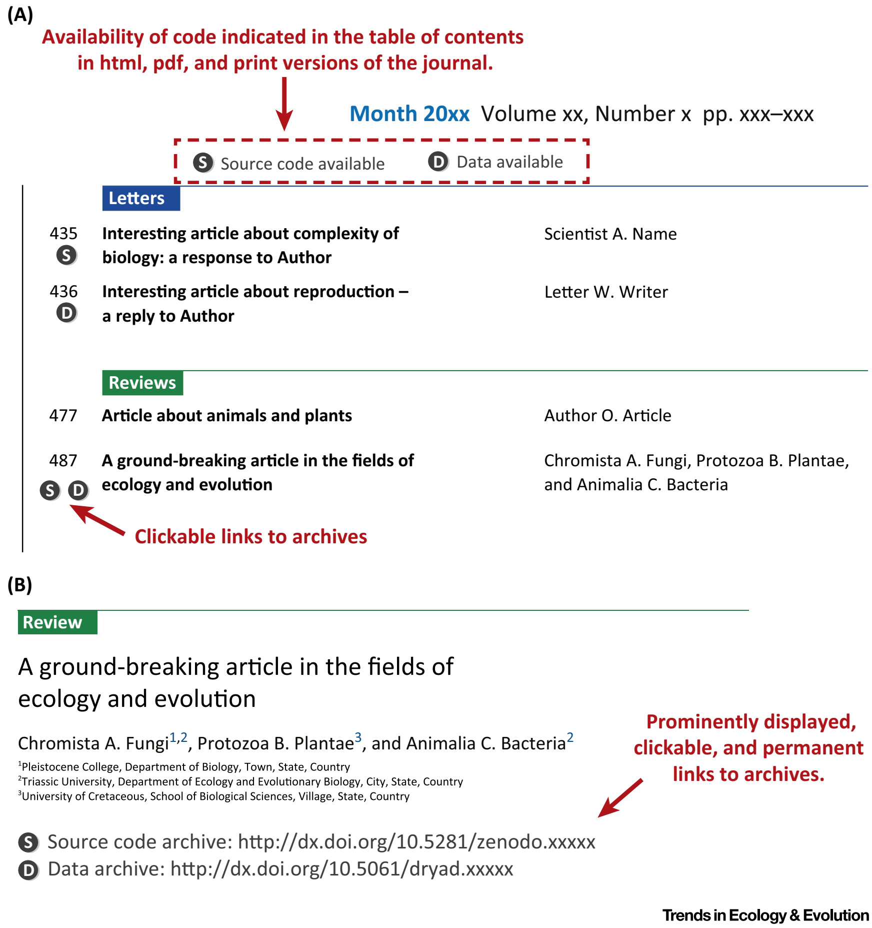


Figure 2. Recommended Journal Page Layouts. (A) For the table of contents of a journal. (B) For the first page of a journal article. The recommendations are for all formats of the journal including html, pdf, and print versions. An important feature is that active links can be clicked in electronic versions to directly access the code. The article titles and author names were made-up for the examples.

## Promoting Code in Ecology

Journals can promote the release of code used in ecological studies by increasing the visibility and discoverability of code and software. One way to increase visibility is to indicate code availability in the table of contents of all formats of the journal and provide direct links from the online table of contents to the code (Figure 2A). In the article, links to code prominently displayed on the first page will also increase visibility (Figure 2B). This article format for data has already been adopted by some ecology journals, including *The American Naturalist*. In addition, journals can require and verify that code is made available at the time an article is submitted for review or is accepted for publication [10]. Requirements by journals for data to be made available have been very successful [3]. Specialized software sections in journals go a step further in promoting highly refined code that can be used broadly for ecological analyses and visualization, and provide an associated publication [11]. Communicating the availability of software in a well-described journal format to the ecology community highlights software as a product of ecological research. Discoverability can be enhanced if searchable databases for articles (e.g., journal archives, Web of Science, and PubMed) include an option for searching for articles with code. This search capability would make it more feasible to find, compare, and adapt code from multiple research articles for a new study. To increase the value of code releases within the existing academic incentive structure, papers and other scientific products that use publicly-available code need to cite the code and associated publication (if there is one). Journals should encourage or require the citing of code, and provide instructions and examples for how to do so in the author instructions. Citing code will increase the impact of journal articles which include code, and provide credit to ecologists developing valuable software resources.

It is also important to consider how best to make ecological code publicly available. Ecologists may not be aware of the steps needed to archive code or the ease of doing so with available resources [3,12,13]. Table 1 compares some of the common resources available for archiving code. A license, which states the conditions under which the code can be used, should be included with a submission to an archive. If a submission does not include a license, then no one will be able to use the code. Most of the resources in Table 1 provide a

Table 1. Comparison of Common Resources (Zenodo, Figshare, Dryad Digital Repository, PANGAEA Data Publisher, GitHub, and Bitbucket) Used for Archiving Code and Data[a]

| | Zenodo | Figshare | Dryad | PANGAEA | GitHub and Bitbucket | Supplementary Material |
|---|---|---|---|---|---|---|
| Default License | Flexible | MIT | CC0 | CC-BY | Flexible | None |
| Long-term | Yes[b] | Yes [b] | Yes [b] | Yes [b] | No | Yes [b] |
| Assigns DOI | Yes | Yes | Yes | Yes | No | No |
| Code Search Option | Yes | Yes | No | No | Yes | No |
| Upload from GitHub | Yes | No | No | No | – | No |
| Cost to Author | None | None | Possible | None | None | None |

[a]For the default licenses: flexible means that multiple license options are available from a menu, MIT is the Massachusetts Institute of Technology License, CC0 is the Creative Commons Zero License, and CC-BY is the Creative Commons Attribution License. DOI, digital object identifier. Zenodo, Figshare, Dryad, and PANGAEA are good options for archiving because they provide licenses, are long-term, and are citable. The cost to authors assumes that the code is publicly available. Note that the information in this table is subject to change.

[b]Long-term availability depends on continued government funding or the success of the companies involved.

license or license options, making it easy to add a license when code is submitted. Archives need to be long-term, assuring continuous availability ([14], https://caseybergman.wordpress.com/2012/11/08/on-the-preservation-of-published-bioinformatics-code-on-github/). All of the resources in Table 1 store submissions for the long-term except for GitHub and Bitbucket. Some of the archives assign code submissions a digital object identifier (DOI), which makes code straightforward to cite in scientific publications. Other considerations are whether it is possible to search specifically for code within the archive, the process for uploading code, and the cost of archiving code. Most of the archives host code for free if the code is made publicly available. Overall, Zenodo, Figshare, Dryad, and PANGAEA are good options for archiving because they provide licenses, are long-term, and are easily citable (Table 1).

Journals can have a significant impact on increasing the value of code within the ecology community. We believe that broad adoption of the suggestions to increase visibility and discoverability of code, require archiving of code, and increase citation incentives for doing so, will motivate more authors to release both analysis code and scientific software. By fostering reproducibility and reuse, more available code can improve the quality and accelerate the rate of research in ecology.


## Acknowledgments

K.A.S. was supported by the Washington Research Foundation Fund for Innovation in Data-Intensive Discovery and the Moore/Sloan Data Science Environments Project at the University of Washington. This work was supported in part by the Gordon and Betty Moore Foundation Data-Driven Discovery Initiative (grants GBMF4553 to J.M.H. and GBMF4563 to E. P.W.). We thank Carl Boettiger for thoughtful comments that significantly improved the paper.



[1]School of Oceanography, University of Washington, 1503 NE Boat Street, Seattle, WA 98105, USA

[2]eScience Institute, University of Washington, 3910 15th Avenue NE, Seattle, WA 98195, USA

[3]Computer Science and Engineering Department, University of Washington, 185 Stevens Way, Seattle, WA 98195, USA

[4]Department of Wildlife Ecology and Conservation, University of Florida, 110 Newins-Ziegler Hall, Gainesville, FL 32611, USA

[5]Informatics Institute, University of Florida, 230 Newell Dr Griffon-Floyd Hall, Gainesville, FL 32611, USA

*Correspondence: kas.mislan@gmail.com (K.A.S. Mislan).


http://dx.doi.org/10.1016/j.tree.2015.11.006


### References

1. Hampton, S.E. *et al.* (2013) Big data and the future of ecology. *Front. Ecol. Environ.* 11, 156–162
2. Peng, R.D. (2011) Reproducible research in computational science. *Science* 334, 1226–1227
3. Hampton, S.E. *et al.* (2015) The Tao of open science for ecology. *Ecosphere* 6, 120
4. Barnes, N. (2010) Publish your computer code: it is good enough. *Nature* 467, 753
5. Rubenstein, M.A. (2012) *Dear Colleague Letter – Issuance of a New NSF Proposal & Award Policies and Procedures Guide,* National Science Foundation www.nsf.gov/pubs/2013/nsf13004/nsf13004.jsp
6. Poisot, T. (2015) Best publishing practices to improve user confidence in scientific software. *Ideas Ecol. Evol.* 8, 50–54
7. Whitlock, M.C. *et al.* (2010) Data archiving. *Am. Nat.* 175, 145–146
8. Stodden, V. *et al.* (2013) Toward reproducible computational research: an empirical analysis of data and code policy adoption by journals. *PLoS ONE* 8, e67111
9. Ince, D.C. *et al.* (2012) The case for open computer programs. *Nature* 482, 485–488
10. Nosek, B.A. *et al.* (2015) Promoting an open research culture. *Science* 348, 1422–1425
11. Pettersson, L.B. and Rahbek, C. (2008) Editorial: launching Software Notes. *Ecography* 31, 3
12. Stodden, V. and Miguez, S. (2014) Best practices for computational science: software infrastructure and environments for reproducible and extensible research. *J. Open Res. Software* 2, 1–6
13. Wilson, G. *et al.* (2014) Best practices for scientific computing. *PLoS Biol.* 12, e1001745
14. White, E.P. (2015) Some thoughts on best publishing practices for scientific software. *Ideas Ecol. Evol.* 8, 55–57


Science & Society

# Communication of Science Advice to Government

Jeffrey A. Hutchings[1,2,3,*] and Nils Chr. Stenseth[2,4,5]

**There are various ways to construct good processes for soliciting and understanding science. Our critique of advisory models finds that a well-supported chief science advisor (CSA) best ensures the**