Increase your impact by sharing papers and source code

Jack W. Baker

May 23, 2021

You should publish papers to document your research and communicate to the world. Publishing is hard work, done with the ultimate goal of having an impact and getting credit for your effort. Writing a rigorous paper that clearly explains your work is the most important thing you can do in this regard. But there are other things you can do to magnify your impact. Specifically, you can publicly share versions of your papers, and you can share your code. I’ll provide some background on these issues below, and explain why I make these recommendations.

Versions of your paper

Let’s start with a few definitions that will help with the discussion below.

**Preprint**  This is a paper that has just been submitted to a journal or is still in preparation. It has not yet undergone peer review (i.e., it has not been revised in response to peer review comments).

**Accepted Manuscript**  This is your version of the manuscript, after you have revised it in response to peer review comments and it has been accepted for publication. It is the last document you submitted to the journal before you got the “Congratulations, your paper is accepted” response. It is also referred to as the *Accepted Version*.

**Published Journal Article**  This is the final version of your accepted paper that the journal will publish, and it includes the journal’s typesetting and formatting. It is also referred to as the *Version of Record*.

Post Preprints

Before you submit your paper to a journal for review, you can post it on a “preprint server” that stores these documents and provides them on the internet. Most journals will review papers that have been posted to preprint servers, with the request that you later update the preprint posting to include a link to the final paper. Some journals now even have systems where you submit once and have the paper simultaneously be posted on a preprint server and start peer review. Lists of which journals allow preprints are at [https://en.wikipedia.org/wiki/List_of_academic_publishers_by_preprint_policy](https://en.wikipedia.org/wiki/List_of_academic_publishers_by_preprint_policy) and [https://v2.sherpa.ac.uk/view/publisher_list/](https://v2.sherpa.ac.uk/view/publisher_list/).

There are several reasons to post preprints:
• **Get feedback.** If you are fortunate, the preprint will receive comments and criticisms from readers. This is great because you will still have time to change it to respond to feedback. If readers find items that you could further improve, wouldn’t you rather know before the paper is finalized?

• **Disseminate quickly and widely.** It can take many months to go through peer review and get to a final publication. While all that is happening, the preprint is already available, so interested users can take advantage of your work earlier. Additionally, preprints are always free for all readers, while journal papers often require expensive subscriptions, so more people can access preprints. Preprints have additional practical benefits: the paper can attract citations earlier, and you can list the preprint on your CV.

• **Stake claim.** Preprints are timestamped and permanent. So if there are any later problems with establishing who first conceived of an idea, preprints provide helpful evidence of the date of your contribution. On two occasions, I have had disputes with others about whether they were taking our groups’ ideas without proper credit. In both cases, we had a public record of prior sharing of our ideas that helped resolve the dispute to our satisfaction.

There is a long history of posting preprints in physics and some other fields. Some people even post papers still in development or that may never be submitted (to share ideas or get feedback). In engineering, this practice has been slower to take off but is increasingly common. At present, there seem to be two good candidates for publication of engineering preprints:

• EngrXiv ([https://en.wikipedia.org/wiki/EngrXiv](https://en.wikipedia.org/wiki/EngrXiv)). This site was founded in 2016 and is run by a nonprofit organization. It is not very active, but the number of papers shared is growing over time.

• Research Square ([https://www.researchsquare.com/browse](https://www.researchsquare.com/browse)). This commercial site allows free posting of preprints and charges for other services to make money. It is much more active than EngrXiv but much broader. It partners with many journals to facilitate the simultaneous posting of preprints when you submit to those journals.

**Post Accepted Manuscripts**

Many journals allow you to post your Accepted Manuscript on your personal webpage for public access (they often require you to include a link to the Published Journal Article as well). You can check at [https://v2.sherpa.ac.uk/romeo/](https://v2.sherpa.ac.uk/romeo/) to see the policies of individual journals. Note that you are usually much more restricted in how you can share the Version of Record. But since the Accepted Manuscript generally has the same information as the Version of Record, it is helpful for you to share that document on your website. This will allow people searching on the internet to find your paper, even if they don’t have access to the journal where your Final Version was published.

**Share source code**

The traditional model of publishing papers is an unsatisfactory way to document computational or data-related scholarship. There are several benefits to publicly sharing the source code and data underlying computational research:
• It more completely documents your research. Readers can understand what was done by looking at and running the code rather than inferring based only on the description in the paper. The Reproducible Research movement argues that you should provide code to document data analysis scholarship for this reason.

• It increases the adoption of your research. Readers are much more likely to adopt a computational approach if they can download and run the code, rather than writing new code from scratch\(^1\).

• It is easier to write the accompanying paper. If you know that you will provide the code, you can write about concepts without worrying about documenting every algorithmic detail.

Services such as GitHub make it easy to share your code. These services also help you develop your code, track bugs, and collaborate.

There are two reasons I see why people are hesitant to share code. First, it is more work to clean up and share your code. But this needn’t be a huge effort, and I think the effort is usually worthwhile. Second, it may be scary to think that someone will use your code to do scholarship you are planning to do. There may be cases where this is reasonable—for example, if you painstakingly gathered data you plan to use in several forthcoming papers, you could wait a while before releasing the data. But more often, the people will use your tools in different ways than you planned to, or they will help you find bugs and improvements. I can’t think of any personal examples where I could not publish on a topic because someone else used my code to do it first.

Once you agree to share code, you have to decide what to provide. I suggest posting a repository that, when run, will reproduce all of the figures and results in your paper; this is ideal, as it fully documents your paper. There are cases when that is not practical (e.g., because you cannot share some data, or you relied on a complex workflow that is difficult to package for re-use). If data privacy limits you from reproducing your figures, consider posting a complete workflow that runs on hypothetical data you made up. Then you have documented the workflow even if readers cannot reproduce your results. If your workflow is too complex to share fully, consider what parts would be helpful to a reader. Is there a particular step in the workflow where your contribution adds a lot of value? In that case, perhaps you can share a nicely packaged tool for only that single step.

When you share, do your best to provide clear and documented code, but remember that you aren’t providing commercial software. Do your best to make it understandable by others, but you don’t need to write a manual or create a user interface. You have already done the reader a big favor by sharing your code, and you can reasonably ask them to do some work to use your code if they are very interested.

DOIs

A Digital Object Identifier (DOI) is a permanent pointer to your paper, data, or code. A DOI looks similar to a Uniform Resource Locator (URL), for example: https://doi.org/10.1038/s41893-020-0508-7. A URL is also a pointer to an online resource, but it is not designed to be permanent. See https://www.doi.org/ for more information.

The DOI system was created in the late 1990s because scientific information was quickly moving to digital architecture, and use snowballed in the 2000s. Now nearly all peer-reviewed journal

\(^1\)The majority of my most highly cited papers have accompanying code that readers can download and use to solve a particular problem. I am certain that these papers would have been much less popular if we had only written a paper and not provided the code.
articles are assigned DOIs. Increasingly, data sets, preprints, and other scholarly documents are also assigned DOIs.

GitHub allows easy assignments of DOIs to code repositories (https://guides.github.com/activities/citable-code/). Doing this has a few benefits. First, authors can then cite a particular version of the code associated with that DOI, so that even if the repository continues to evolve, it is clear what code was used by a citing document. Second, the DOI links are designed to be stable over time, unlike basic URL links. Third, if you are the code author, creating a DOI allows others to cite your code and lets you get credit for citations.

Other services like https://zenodo.org/ and https://www.designsafe-ci.org/data/browser/public/ allow you to upload data sets and create a DOI as well. With so many easy-to-use services available, and multiple benefits of creating DOIs, it is a good practice to create a DOI for any scholarly information that you anticipate others would find useful.

Summary recommendations

Given the above, I recommend making the following steps part of your plans to publish papers about your research. The incremental effort to do these things is small compared to the work of actually doing the research, and they should increase the impact of that research.

- Publish in journals that allow posting of Preprints and Accepted Manuscripts.
- Post a Preprint of your manuscript when you submit it to a journal.
- Share source code to document your work and create a DOI for it.
- Share your Accepted Manuscripts on your personal website.