How to share + cite research code (and get credit)

Slides for the Stanford/UCSF Open Science Journal Club

Ariel Deardorff, MLIS
Data Services Librarian, UCSF Library

UCSF Library

Today's Agenda

- Short intro: Sharing research code: why and how? (20 min)
- Breakout discussion #1 (15 min)
- Breakout discussion #2 (15 min)
- Group wrap up (10 min)

Why talk about code and research software?

For many of us science wouldn't happen without code and software!

"Software is a cornerstone of science. Without software, twenty-first century science would be impossible. Without better software, science cannot progress." - Science Code Manifesto

We should **share** and **cite** the code we use in our research to:

- Give and receive credit for creating software
- Help others discover useful software
- Make our research more reproducible

Open Code in two steps:

1. Share the code that you write
2. Cite the code and software that you use

Caveat: code sharing and citation practices are evolving! These slides were informed by the Software Citation Principles and Recognizing the Value of Software: A Software Citation Guide

However, more publishers are realizing the importance of citing code and coming up with new author guidelines (ex: *Science* asks authors to cite software central to their findings).

How do you share your code?

Scenario 1: You wrote some code for your data analysis

- Deposit your code in an archival repository like Zenodo, get a DOI, and link to your paper
    - Why is GitHub not good enough? Use the GitHub to Zenodo pipeline!

Scenario 2: You created your own piece of software

- Create a software package and submit it to a disciplinary or programming language-specific repository
    - Ex: for an R package you could submit to CRAN or Bioconductor
- Consider writing a software paper about your package and submitting it to a journal like the *Journal of Open Source Software*

How do you cite code?

Scenario 1: You are using someone's software in your analysis

- Find their citation file or assemble it yourself
  - For R packages: citation("packagename")
  - Citation guide for common Python packages
- Cite it in the references of your paper (note the version!)

Scenario 2: You are using someone's software to write new software

- Find their citation file
- Cite it in your source code/documentation

Note from Katz et al., 2021: If an article exists that describes the software, it should be cited as an additional reference, as well as citing the software itself. Do not cite the article instead of the software.

What to include in a citation?

Developer, A. A., Developer, B. B., & Developer, C. C. (yyyy)[1]. *Title of the software: Subtitle* (Version #.#)[2] [Computer software][3]. Publisher[4], https://URL[5]

Coon, E., Berndt, M., Jan, A., Svyatsky, D., Atchley, A., Kikinzon, E., Harp, D., Manzini, G., Shelef, E., Lipnikov, K., Garimella, R., Xu, C., Moulton, D., Karra, S., Painter, S., Jafarov, E., & Molins, S. (2020, March 25). *Advanced Terrestrial Simulator (ATS) v0.88* (Version 0.88) [Computer software]. Zenodo. https://doi.org/10.5281/zenodo.3727209

Dataverse Project (2020). *Dataverse* (Version 4.20) [Computer software] https://github.com/IQSS/dataverse/releases/tag/v4.20

IBM Corp. (2017). *IBM SPSS Statistics for Windows* (Version 25.0) [Computer software]. IBM Corp. https://www.ibm.com/products/spss-statistics

Source: Katz et al., 2021

What code should you cite?

This is an evolving area and can depend on your community norms. However, in general:

“software should be cited on the same basis as any other research product such as a paper or a book; that is, authors should cite the appropriate set of software products just as they cite the appropriate set of papers.”

- Software Citation Principles

Think about the main tools or packages that you used in your work

While we are talking about code...

Version control your scripts!

Comment everywhere!

Other useful guidance:

- Good Enough Practices in Scientific Computing
- Excuse me, do you have a minute to talk about version control?

Reminder: Get an ORCID!

ORCID = researcher ID

An ORCID helps make sure that you (not someone with a very similar name) is getting the credit for your work

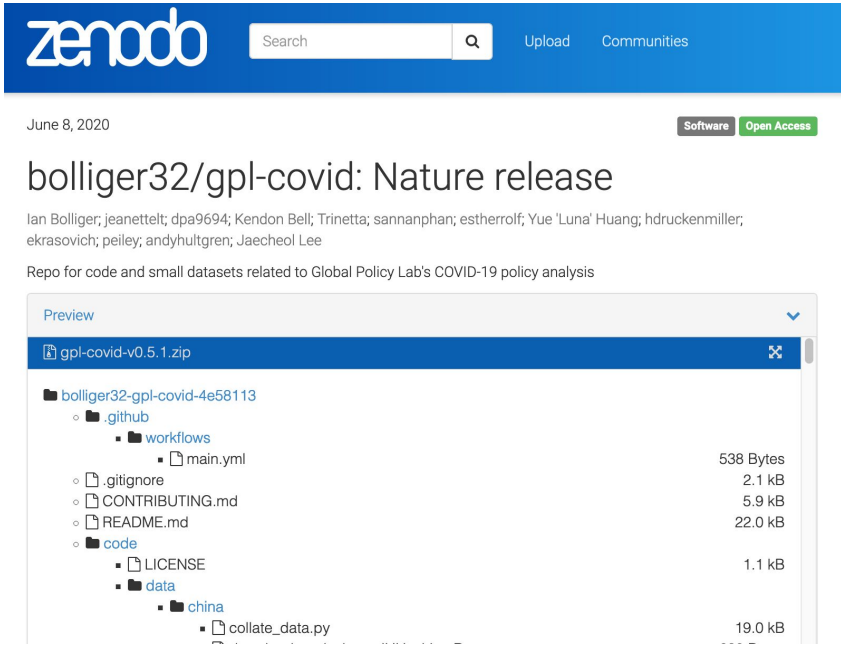
Breakout Discussion #1

1. How do you use code and software in your research?
2. Do you cite code? Is there a standard in your field?
3. Do you share or publish scripts or code that you have written?

Breakout Discussion #2

Explore this code repo in Zenodo.

- What is this code for?
- What is it linked to?
- How would you cite it?



What was it for?

```
![build](https://github.com/bolliger32/gpl-covid/workflows/CI/badge.svg)
[![License: MIT](https://img.shields.io/badge/License-MIT-yellow.svg)](https://opensource.org/licenses/MIT)
[![DOI](https://zenodo.org/badge/DOI/10.5281/zenodo.3832367.svg)](https://doi.org/10.5281/zenodo.3832367)
# The Effect of Large-Scale Anti-Contagion Policies on the COVID-19 Pandemic

This repository contains code and data necessary to replicate the findings of [Hsiang et al. (Nature 2020)](https://www.nature.com/articles/s41586-020-2404-8).

## Setup
Scripts in this repository are written in R, Python, and Stata. Note that you will need a Stata license to fully replicate the analysis (provided in the CodeOcean capsule). Throughout this Readme, when indicating paths to code and data, it is assumed that you'll execute scripts from the repo root directory.

### CodeOcean Capsule
The easiest way to interact with our code and data is via our [CodeOcean capsule](https://codeocean.com/capsule/1887579/tree/v2), because all of the relevant setup described below has been done for you. You may replicate the full analysis through the "Reproducible Run" feature or interact directly with our code through Jupyter Notebooks that run Python, R, and Stata. You may also utilize RStudio. If you wish to use the command line on a cloud workstation, you will want to activate our conda environment with `conda activate gpl-covid`.

### Github Repository
You may also view and download source code from our [Github Repository](https://github.com/bolliger32/gpl-covid). "v0.5.1" is the tag that is associated with the current version of the manuscript (as of 06/07/2020), but you may also view the latest codebase and datasets on the master branch. To run this code, you will first want to create and activate our [conda](https://docs.conda.io/projects/conda/en/latest/index.html) environment.

```bash
```
```

Article

The effect of large-scale anti-contagion policies on the COVID-19 pandemic

https://doi.org/10.1038/s41586-020-2404-8

Received: 22 March 2020

Accepted: 26 May 2020

Published online: 8 June 2020

Solomon Hsiang[1,2]✉, Daniel Allen[1], Sébastien Annan-Phan[1,3], Kendon Bell[1,4], Ian Bolliger[1,5], Trinetta Chong[1], Hannah Druckenmiller[1,3], Luna Yue Huang[1,3], Andrew Hultgren[1,3], Emma Krasovich[1], Peiley Lau[1,3], Jaecheol Lee[1,3], Esther Rolf[1,6], Jeanette Tseng[1] & Tiffany Wu[1]

Governments around the world are responding to the novel coronavirus (COVID-19) pandemic[1] with unprecedented policies designed to slow the growth rate of infections. Many actions, such as closing schools and restricting populations to their homes, impose large and visible costs on society, but their benefits cannot be directly observed and are currently understood only through process-based simulations[2–4]. Here, we compile new data on 1,717 local, regional, and national non-pharmaceutical interventions deployed in the ongoing pandemic across localities in China, South Korea, Italy, Iran, France, and the United States (US). We then apply reduced-form econometric methods, commonly used to measure the effect of policies on economic growth[5,6], to empirically evaluate the effect that these anti-contagion policies have had on the growth rate of infections. In the absence of policy actions, we estimate that early infections of COVID-19 exhibit exponential growth rates of roughly 38% per day. We find that anti-contagion policies have significantly and substantially slowed this growth. Some policies have different impacts on different populations, but we obtain

Lots of things in lots of places!

Data availability

The datasets generated and/or analysed during the current study are available at https://github.com/bolliger32/gpl-covid. Future updates and/or extensions to data or code will be listed at http://www.globalpolicy.science/covid19.

Code availability

For easier replication, we have created a CodeOcean 'capsule', which contains a pre-built computing environment in addition to the source code and data. This is available at https://codeocean.com/capsule/1887579/tree/v1. Future updates and/or extensions to data or code will be listed at http://www.globalpolicy.science/covid19.

How would you cite this code?

Cite as

Ian Bolliger, jeanettelt, dpa9694, Kendon Bell, Trinetta, sannanphan, ... Jaecheol Lee. (2020, June 8). bolliger32/gpl-covid: Nature release (Version v0.5.1). Zenodo. http://doi.org/10.5281/zenodo.3885328

What other questions/comments do you have?

Next Meetup:
Feb 23rd - Open Data Standards