



Setup

[Edit](#) [New page](#)[Jump to bottom](#)

Matthew Gentzkow edited this page on Jun 5, 2020 · 17 revisions

The following are the core tools we use in our projects. Individual Github repositories document their own requirements, including specific modules or libraries as well as software not listed here.

We recommend that anyone who will be working on our projects take the time to become familiar with Github and Git, as well as any of the other tools they will be using actively. We expect RAs to be proficient in all of the core tools.

Github & Git

We store our project repositories on Github. We use the Git version control system to organize our code and data. We use issues on Github to manage tasks and structure communication around projects. We use Github wikis (like this one!) to store supplemental information related to our projects.

Anyone who is new to Git / Github should start by reading the following Github guides (takes about 20 minutes).

- [Git Handbook](#)
- [Understanding the GitHub Flow](#)
- [Mastering Issues](#)
- [Mastering Markdown](#)

The book [Pro Git](#) is a standard source of more detailed documentation. It provides a good practical reference for how to execute specific tasks. It is freely available online.

For deeper understanding, we recommend chapters 4-9 of the book [Version Control with Git](#). It provides a more detailed ground up explanation of how Git works and why. Many aspects of Git are confusing to newcomers, and understanding the underlying structure will make your work more efficient (and less frustrating!).

Setup Steps: [Create a Github account and install the Git desktop / command line clients](#). Give your Github ID to a lab member who can give you permissions to the appropriate repositories.

Git Large File Storage (LFS)

Git LFS is a separate piece of software that allows Git to handle large files. We require everyone running one of our repositories to have Git LFS installed because inadvertently committing large files directly can cause bad things to happen.

Setup Steps: [Install Git LFS](#). Note that you only need to do step 1 under "Getting Started" at this point.

Dropbox

We use Dropbox to store files that are too big for Github (even with LFS), files that need to be shared across multiple projects, archives from pre-Github projects, and other kinds of files that don't have a natural home in repositories.

Setup Steps: [Install Dropbox](#). Ask a lab member to give you access to the any Dropbox directories you will need for your project(s).

Python

We use Python to control the running of code and for many other data building and analysis tasks. There are many excellent online introductions to Python including [this one](#) from Software Carpentry. The book [Learning Python](#) by Mark Lutz is a definitive manual. [Datacamp](#) is a very useful set of interactive courses focusing specifically on data analysis; several introductory courses are free, while more advanced courses require a subscription.

Python currently exists in two different development streams, Python 2 (versions numbered 2.X) and Python 3 (versions numbered 3.X). We use Python 3 for our projects and you should make sure you have this version installed.

We have developed some simple Python tools for managing the execution of code in our repositories which are collected in the [gslab_make](#) repository. You can learn more about these tools as you begin working with our repositories.

Setup Steps: All Mac / Linux machines and many Windows machines come with Python installed. You can confirm whether it is by going to a terminal / bash window and typing `python`. If it is installed, you should see a welcome message indicating the version of Python (you can type `quit()` to exit). If it is not installed, or if you only have the 2.X version, you can install the most recent version [here](#).

LyX

We use LyX as our front end LaTeX editor for most projects. Documentation is on the main page [here](#). Note that you will need to install a TeX system such as MiKTeX on Windows or MacTeX on Mac OS before installing the LyX software itself. Instructions for this are on the LyX download page.

Setup Steps: [Install LyX](#)

R

R is one of the main statistical packages we use. Good resources for learning R are the [Analysis](#) and [Programming](#) tutorials from Software Carpentry. [R4DS](#) is a great introduction to all stages of the data analysis pipeline. [Datacamp](#) is a very useful set of interactive courses covering a wide range of topics; several introductory courses are free, while more advanced courses require a subscription.

Setup Steps: [Install R](#). [Install R Studio](#), a useful integrated development environment.

Stata

[Stata](#) is the other main statistical package we use. Some resources for learning Stata are the [UNC Population Center Tutorial](#) and Christopher Baum's lecture [A Little Bit of Stata Programming Goes a Long Way](#).

Setup Steps: Install Stata. Stata is commercial software and will be typically purchased / installed through your university.

+ Add a custom footer

▸ Pages 28

Lab Manual

- [Setup](#)
- [Workflow](#)
 - [Details](#)
- [Code](#)
- [Data](#)
- [Papers & Slides](#)
- [Production](#)
- [Releases](#)

Appendices

- [Practice Task](#)
- [Autofilling Values](#)
- [IT Support](#)
- [Research Clusters](#)
- [Legacy Tools](#)
- [Style Guides](#)
 - [Papers](#)
 - [Python](#)
 - [R](#)
 - [Stata](#)
 - [C](#)
- [Mothballing Projects](#)
- [Recruiting on Social Media](#)
- [PhD Applications](#)
- [Gentzkow-Shapiro Lab Notes](#)
 - [Lab Admin](#)
 - [Dropbox](#)
- [Allcott-Gentzkow Lab Notes](#)
 - [Meeting Notes](#)
- [JMSLab Notes](#)

Clone this wiki locally

<https://github.com/gslab-econ/lab-manual.wiki.git>

