

Math 355: Math for Machine Learning

Fall 2019

Branden Stone

Introduction to GitHub

Getting started with Git

Today we will discuss the delivery method of all course materials, assignments, and projects. This method will rely on learning `git`.

- According to Wikipedia, “Git is a version control system that is used for software development and other version control tasks. As a distributed revision control system it is aimed at speed, data integrity, and support for distributed, non-linear workflows.”
- Basically, it is a more sophisticated Dropbox and allows you to keep your code up to date across multiple platforms.
- A complete walk through of Git can be found online at <https://git-scm.com/book/en/v2>.
- Today we will only learn the basics.

Sign up with GitHub

If you don’t already have an account, you will need a GitHub account. You can do so here: <https://github.com/>.



- I'll wait here while you do this.
- Once you are finished, come up and enter your username on my computer, this send out an email inviting you to join the class.

Welcome to the class!

- Once you have accepted your invite, you now have access to the courses materials.
- <https://github.com/math-355>
- The page should look something like this. Feel free to poke around.

The screenshot shows a GitHub repository page for 'Math for Machine Learning'. The repository is owned by 'Hamilton' and is described as 'The course files for Math 355'. The email address 'bstone@hamilton.edu' is listed. The repository is public and has a 'Private' label. The repository is categorized as 'HTML' and has 0 stars, 0 forks, and 0 watchers. It was updated 19 hours ago. The repository contains two files: 'Week01-Intro-to-Git-and-ML' and 'Fall-2019-Syllabus'. The 'Fall-2019-Syllabus' file is also labeled 'Private' and has 0 stars, 0 forks, and 0 watchers. It was updated 27 days ago. On the right side of the page, there are sections for 'Top languages' (HTML) and 'People' (1 person). There is a button to 'Invite someone'.

Mandatory Syllabus time.

You can find the syllabus as well as resources on the landing page <https://github.com/math-355>

Cloning Forking a Repo

First, what is a repo and why would you want to do these things to it? **Repo** is short for repository which is basically a directory of files and folders. Think of your GitHub account as a large folder containing other folders (repos). Each repo is a different project, or in this case, a weekly objective. If you just created a GitHub account, then you don't have any repos yet. We are going to change that.

This is where forking comes in. You need to copy the **Lab01-Intro-to-GitHub-Spring-2018** repo into your account. This way you can edit/add files. A **Fork** of a repo is like a fork in a road. There is a main road (the original **Lab01-Intro-to-GitHub-Spring-2018** repo) and a fork off the road (your repo). At any time you are editing the forked repo (or traveling the forked road), you can go back to the original.

Forking the a repo is super easy. All you need to do is click the fork button at the top of this page (<https://github.com/csc-171/Lab01-Intro-to-GitHub-Spring-2018>) and select the account to fork to. **Do this now, then scroll down to this spot on the new page.**

Bam! You have forked a repo! Congratulations. If all went well, you should see something like this at the top of the page:

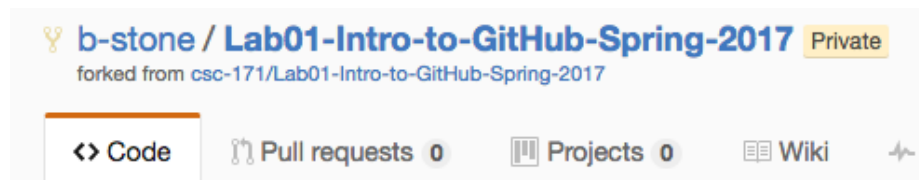


Figure 1: image

The only difference is that your username should be displayed and not **b-stone**. I.e. you should see something like this, **YOURUSERNAME/Lab01-Intro-to-GitHub-Spring-2018**.

From now on, this is the first thing you should do when you start a lab in this class.

So now what? You have a forked repo sitting on some server. How do you get your hands on it? Well you could use the GitHub interface, but that is lame. So we need to get the files to your local machine. There are several ways to do this, but the cool kids like to clone. **Cloning** is just copying and pasting the repo to

your local machine. To do this we need to get our hands a little dirty with the following steps.

- Open **Git Bash** (or **Git Shell** on some machines) on your machine. This is a command line interface that allows you to communicate with GitHub.
- Once you have **Git Bash** open, type **cd Desktop** and hit enter. This will make sure all downloads are placed on your desktop.
- Look for a green button on this page that says **Clone or download**, click it, and copy the text. It should look something like this:
`https://github.com/YOURUSERNAME/Lab01-Intro-to-GitHub-Spring-2018.git`.
- Go back to Git Bash and enter the following (note, don't enter the \$, it's just there to tell you that is the start of the line):

```
$ git clone https://github.com/YOURUSERNAME/Lab01-Intro-to-GitHub-Spring-2018.git
```

- Next change directory to the repo by typing:

```
$ cd Lab01-Intro-to-GitHub-Spring-2018
```

- Look into the repo by typing:

```
$ ls
```

You see? All of the files are there! This is great right? Now you have a local copy of the repo sitting on your machine. No matter what you do, you will not mess up the originals. All of your edits/changes will be local on your machine until you push your changes back to the server. So let's do that.

Why should I care about Git?

The main reason right now is because we will be using it to turn in assignments during this class. But besides that it is a very powerful tool that will help future you be successful. Here are a few reasons why git is great:

- Version control, keeps all machines up to date with latest code
- Helps keep track of conflicts on collaborative projects
- Complete history of your projects
- Easy testing without breaking your code

On top of this, git manages all of your files for you. But to do this it has to know your file is there. Hence you have to add new files.

Adding Files

Let's do some text editing. So all of us are on the same page, download and install <https://atom.io/>. Once installed, open up this lab repo. Atom is just another text editor. Throughout this semester we will be exposed to a variety of editors. At some point you just need to pick your favorite.

Let's create a new file. Do the following:

- Use Atom to create a new file in ~/Lab01-Intro-to-GitHub-Spring-2018/ called MyFirstJavaCode.java.
- Cut and paste the following into the file:

```
public class MyFirstJavaCode {
    public static void main(String[] args) {
        // Prints whatever you want in the terminal window.
        System.out.println("Hello, World");
    }
}
```

- Change "Hello, World" to whatever you want. Make sure and keep the quotes. Save this file.
- At this point Git does not even know your file exists. To see what I mean run the following in Git Bash:

```
$ git status
```

You see? Untracked files! In order for Git to track your files, you need to add it. So let's do that:

```
$ git add MyFirstJavaCode.java
```

Typing `git status` again should show you that there is a new file being tracked.

Now you can edit to your hearts content and Git will track all of your changes. But how are the changes tracked?

Committing Changes

```
$ git commit -am 'message that describes the change'
```

Pulling (Pulling down server info to local machine)

```
$ git pull
```

Pushing (Pushing up local changes to server)

```
$ git push
```

Pull Request (Submitting Lab Assignments)

Using the GitHub interface.

Review

Here is a quick overview of todays lab. For all future labs, the work-flow will be as follows:

- Fork repo to your account if not already there

- Clone repo to your local machine (`git clone url/to/repo`)
- Navigate to repo (using `cd` and `ls`)
- Add files if needed (`git add filename`)
- Commit any new changes as you go (`git commit -a -m 'message'`)
- Pull new changes from server (`git pull`)
- Push your changes to server (`git push`)
- Submit assignment with pull request (using GitHub interface)

Intro to Machine Learning

What is it?