

# GitHub Cheat Sheet

Econ 213R - Applied Machine Learning

## What is GitHub?

- Explain idea of GitHub
- Define “repository”
- Define “fork”, “clone”, “commit”, “push/pull”
- Define “branch” (master and other branches you can make)
- Define push/pull
- Explain what a README is

## Register for GitHub

First, you need to make an account with GitHub. Go to [www.github.com](https://www.github.com). Sign up from the home page with any email and username you want. Follow the initial signup steps. Note that private repositories require monthly payments; however, you can have an unlimited number of public repositories for free. Once you are done signing up, you are ready to create and fork repositories.

## Creating Repositories

Navigate to your GitHub profile. It should look something like Figure 1.

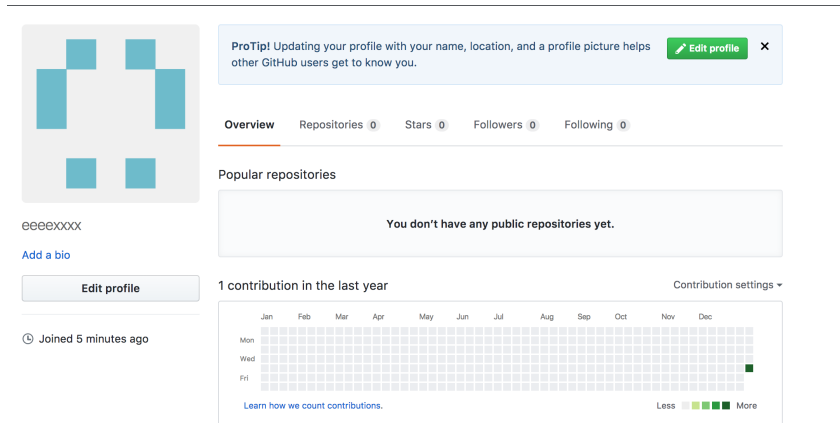


Figure 1: Profile Screen

Click the “Repositories” tab. Click the button that says “New” (see Figure 2).

Name your repository. It is customary to use dashes instead of underscores in repository names. Make sure to initialize the repository with a README.md (see Figure 3). If you accidentally forget to do this, that’s okay; you can always add a README later.

Click “Create Repository” at the bottom of the page. You should see a page that looks like Figure 4.

Congratulations! You have just created your first GitHub repository.

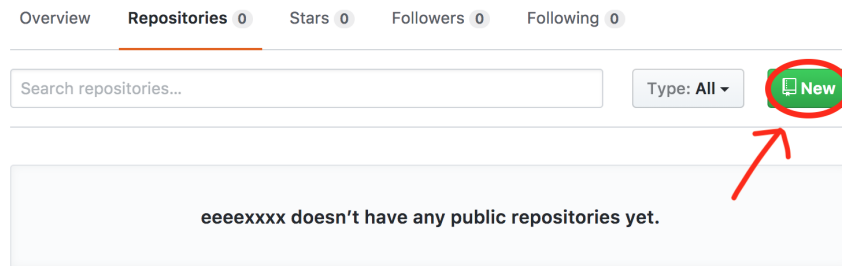


Figure 2: New Repository

## Create a new repository

A repository contains all the files for your project, including the revision history.

**Owner** eeeexxxx / **Repository name** example-repo ✓

Great repository names are short and memorable. Need inspiration? How about [musical-adventure](#).

**Description (optional)**

☒ **Public**  
Anyone can see this repository. You choose who can commit.

☐ **Private**  
You choose who can see and commit to this repository.

☒ **Initialize this repository with a README**  
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None ⓘ

**Create repository**

Figure 3: Initialize the repository.

## Making Changes to Repositories

There are a few ways to upload and change files in a GitHub repository. The most common way to modify repositories is via the command line; see the next subsection for command line instructions. It is highly, *highly* recommended that you learn the command line to modify repositories. The command line allows you to more freedom to modify your repositories and branches, and can streamline your workflow. However, you can use the GitHub website to upload and download files to and from repositories. You can also modify files using the website. Note that you can't run Jupyter notebooks on the GitHub website, and when you try to edit Jupyter notebooks on the website, you have to comb through lots of nasty JSON lingo. If you choose to use the website to make changes to the GitHub repository, you will have to download the Jupyter notebook, modify it on a local machine, and upload it back to the repository. Instructions for using the website are in the subsection following the command line instructions.

If you do not have Git installed on your local machine, use the links below to download Git.

Macs: <https://sourceforge.net/projects/git-osx-installer/files/>

Windows: <http://gitforwindows.org/>

Linux: In the Terminal, first type `sudo apt-get update` and hit Enter. Then type `sudo`

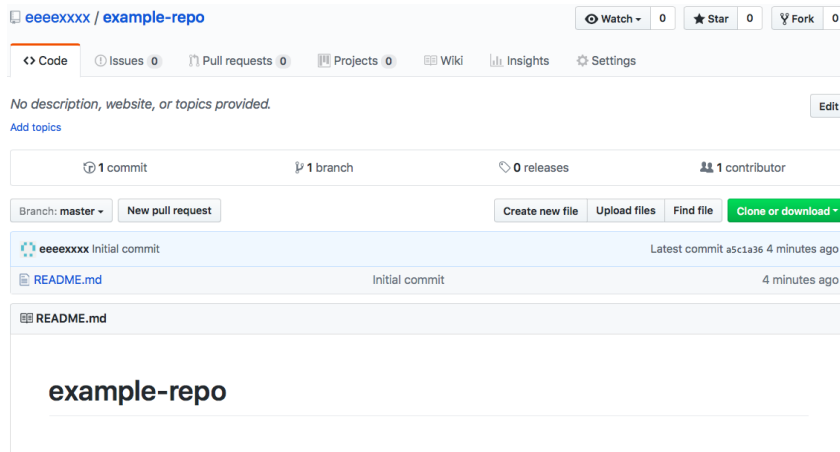


Figure 4: Repository screen.

`apt-get install git` and hit Enter.

## Using Command Line

Let's walk through an example together. In this example, we will configure Git on a local machine, clone the repository to a local machine, create a Jupyter notebook, add it to the repository, and push the changes.

First, set your username and email in Git. Use the commands below to do this.

```
git config --global user.name "Your Name Here"
```

```
git config --global user.email "your_email@website.com"
```

Now Git is configured on your machine, and you are ready to clone the repository you made. First, go to the repository screen on the GitHub website (Figure 4). Click the button that says “Clone or download”. A drop-down window should appear that contains a link to your repository (see Figure 5). Copy the link in the drop-down window.

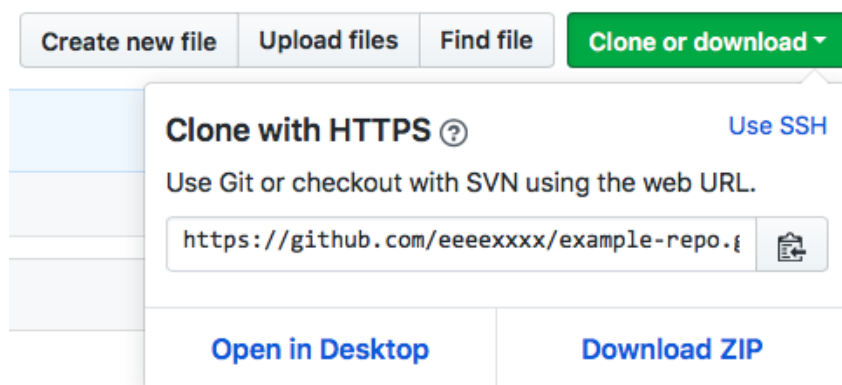


Figure 5: Jupyter notebook screen.

Now open up your command line. Navigate to the location where you want your repository to be (for example, I store all of my files on my desktop, so I would navigate to my desktop from my command line). Once you have navigated to the location, type the following command: `git clone link.to.your.repo`. When your computer is done downloading the repository, you should see a folder with the name of your repository in the location where you wanted it. Now you are ready to make some changes and push them to your master branch.

## Making Changes

For this example, we are going to make a Jupyter notebook and push it to the master branch. In the command line, navigate to your repository, if you're not there already. Type `jupyter notebook` and hit enter. A web browser should open up with a page that looks like Figure 6. If a web browser did not automatically open, the command line should have provided a URL that you can copy and past directly into a browser to access the Jupyter notebook.

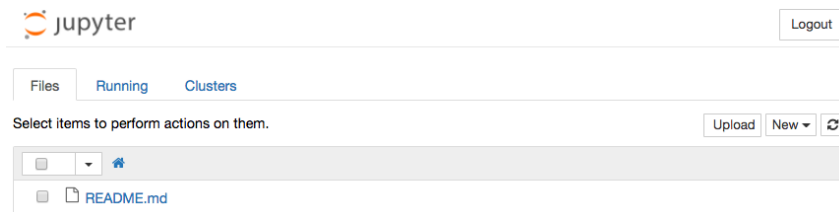


Figure 6: Jupyter notebook main screen.

In the upper right-hand corner, click “New”. A dropdown menu will appear. Under the “Notebooks” heading, click on Python 3 (see Figure 7). When you click “Python 3”, a new window will open in your browser with a screen that looks like Figure 8.

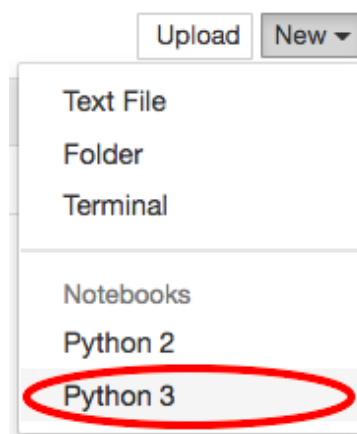


Figure 7: Create a Python 3 Jupyter notebook.

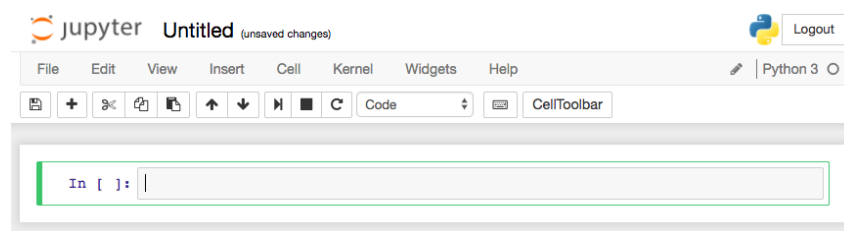


Figure 8: A Jupyter notebook.

Change the name of the notebook from “Untitled” to “example”. To do this, click the “Untitled” text at the top of the notebook. A window will appear; replace the “Untitled” text with “example”. See Figure 9.

In the first cell, type `print("Hello World")`. Then press Shift+Enter. This key combination runs the cell. A new cell will automatically appear below the cell with the print statement. To save the notebook, hit Command+S. Your notebook should now look like Figure 10.

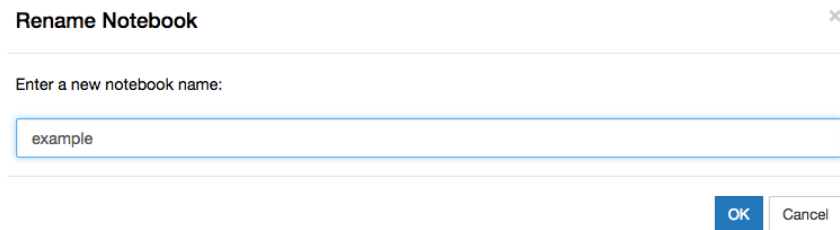


Figure 9: Rename the notebook.

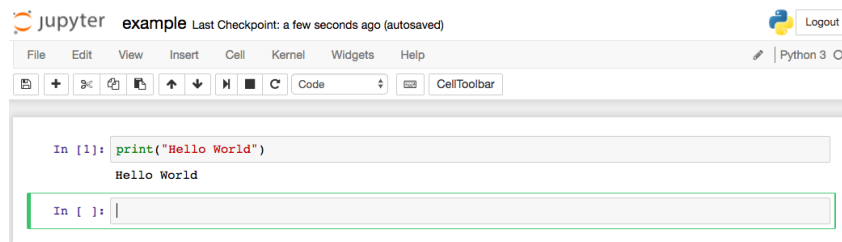


Figure 10: A Jupyter notebook.

Now navigate to the command line where you launched Jupyter. To stop the Jupyter server from running, hit `Ctrl+c`. You will be prompted with the following message: **Shutdown this notebook server (y/[n])?** . Type “y” and hit enter. This will shut down the Jupyter notebook. If you didn’t exit the browser window yet, it will look like Figure 11 after you shut down the Jupyter server. You are safe to close all browser windows related to this Jupyter notebook.

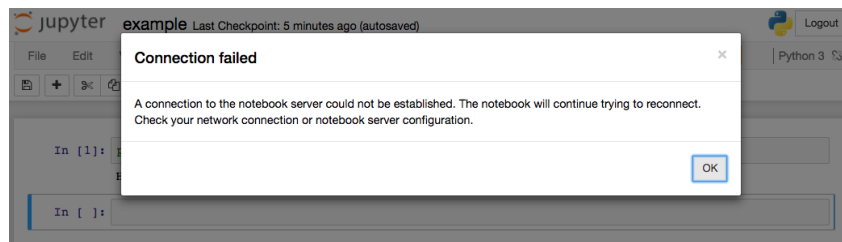


Figure 11: Shutting down a Jupyter notebook.

## Committing Changes

In the command prompt, navigate to the repository, if you aren’t already there. Type `git status` and hit enter. You should see a message that looks similar to Figure ??.

You want to add the notebook you just created to the Git index so Git knows to keep track of this file and any future changes made to it. To do this, type `git add example.ipynb` and hit enter. Now we are going to commit this change to the branch. Then, type `git commit -m "example notebook"`. A message that looks like Figure 13 will appear. Note that you can type any descriptive message you want in the quotes. These messages are to help you keep track of the changes you make in each commit. Now we want to push these changes up the master branch, so that any other machines with this repository can pull the changes you made. Type `git push origin master` and hit enter.

```
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        .ipynb_checkpoints/
        example.ipynb

nothing added to commit but untracked files present (use "git add" to track)
```

Figure 12: git status

```
[master 7353846] example notebook
1 file changed, 53 insertions(+)
create mode 100644 example.ipynb
```

Figure 13: git commit

## Pulling Changes

## Pull Requests

## Using the Website

## Downloading Files

## Uploading Files

## Forking Repositories

Navigate to the class repository. The link is [https://github.com/tfolkman/byu\\_econ\\_applied\\_machine\\_learning](https://github.com/tfolkman/byu_econ_applied_machine_learning).

## Deleting Repositories

PROCEED WITH CAUTION. This action cannot be undone.