SDSE Lab 1: Setting up the environment

Due date: February 10th, 2022

Part 1. GitHub and Git

1. Some of the code we will use will be distributed via GitHub. If you do not already have a GitHub account, please get one at `https://github.com/`.
2. Git is a version control language that is essential for programmers. It's a good idea to know the basics of git, and you can learn that by reading the first 3 chapters of the Git Pro book, freely available here: `https://git-scm.com/book/en/v2`. I believe that Mac/Unix machines come with git installed, and accessible through the command line. There is also a command line option for Windows: `https://git-scm.com/download/win`. There are several GUI-based interfaces, if you prefer that. See `https://git-scm.com/downloads/guis`, as well as GitHub's desktop client: `https://desktop.github.com/`.
3. Visit `https://github.com/ggomes/sdse_public`. This is where you will find any code or data related to homeworks and labs. We will now connect your computer to this repo. Although it is not strictly necessary to use git for this (we could just download the repo in a Zip file), we will do it this way to demonstrate git. It will also be easier this way to stay up-to-date with changes to the repo.
   (a) Click on the green "Code' button. Select on HTTPS and copy the text string. This is the address of the repo that you will provide to your git client to clone the repo.
   (b) But before doing this, you should navigate to a folder where you would like to place this repo. The 'git clone' command will copy the entire repo to your computer and place it in a folder called 'sdse_public'.
   (c) Clone the repo. For example if you are using command line git, you will type:

   ```
   git clone https://github.com/ggomes/sdse_public
   ```

   GUI clients will have different methods for cloning the repo.

Part 2. Install Anaconda or Miniconda

Anaconda is a virtual environment and package manager for Python. It is just as good as Python's native virtual environment and package manager (venv and pip), it just happens to be the one I use.

Here is the website: `https://www.anaconda.com/`

Anaconda is often referred to simply as "conda". There is a free "Individual Version" of conda that comes with a) many packages pre-installed, and b) a GUI interface. Find it here: `https://www.anaconda.com/products/individual`.

Feel free to install Anaconda Individual if you prefer to work with the GUI. However we will only be using a couple commands here, so I would recommend the strip-down version of conda called "miniconda", which is much smaller than the full Anaconda program. You can find Miniconda installers for Windows, Mac, and Linux here: `https://docs.conda.io/en/latest/miniconda.html`.

Part 3. Create the conda environment for the course

These steps are also listed on the GitHub page.

1. Create the ‘sdse’ environment. Note: This is a single command, there should be no line break.

   ```
   conda create -n sdse -c conda-forge python=3.9 jupyter numpy matplotlib seaborn pandas
   scipy dill scikit-learn keras jupytext sympy
   ```

2. Activate the environment. Note that you will have to activate the sdse environment every time you work on code for this course.

   ```
   conda activate sdse
   ```

3. Navigate to your coding folder

   ```
   cd <folder name>
   ```

4. Launch Jupyter

   ```
   jupyter notebook
   ```

Part 4. Getting new data and code. Converting py to ipynb.

I will be adding new material to the GitHub repo as we move through the semester. To obtain that material, you need to ‘pull’ the latest updates from GitHub (or re-download the repo as a zip file). Here are the steps if you are using command-line git.

1. Navigate to the sdse_public folder.
2. Enter this command:

   ```
   git pull
   ```

3. Jupyter notebooks (ipynb files) are stored in the repo as raw Python files. Use ‘jupytext’ to convert them back into notebooks. The script for doing this is called ‘’nb2py.py’

   ```
   python nb2py.py
   ```