Setting up Your Python Environment

Created for the UM Computational Social Sciences Workshop in January 2017

Overview

- Installing Anaconda (python)
  - Lab computers how-to
  - Other computers: docs.continuum.io/anaconda/install
  - Aside: Why Anaconda?
- Downloading Lab Materials
  - CSS GitHub: github.com/UM-CSS
  - How to get files from github
- Jupyter Notebooks
  - How to use them
  - Aside: why they're pretty great for data people

Installing Anaconda (on Lab Computers)

1. Go to the Desktop
2. Double click the Anaconda icon to install
3. Choose "install for me only"

Possible problems (call someone over to help):

- No Anaconda icon
- Error: "Could not create ____"

Installing Anaconda on another Computer

- Download Anaconda
  - www.continuum.io/downloads
  - If you're new to python, choose version 3.
    - If you've been working in 2.7, you might not want to change yet. See here: wiki.python.org/moin/Python2orPython3
- Follow the detailed instructions for Windows, Mac, or Linux on their website:
  - docs.continuum.io/anaconda/install

Aside: Why Anaconda?

Anaconda is a sort of package that contains everything you need to run python. There are other ways to install python and its various libraries and utilities, but anaconda ("conda" for short) has some advantages:

- It comes with all the major data analysis libraries automatically included
- It lets you install additional libraries as you need them.
- It guarantees everything "plays nice" together (i.e. no more "I installed that but my code can't find it" problems or issues with dependency versions being incompatible.)
- It is very up-to-date (unlike many operating system repositories available through yum, apt-get, or dnf)
- In short, you install anaconda and everything just works, painlessly.
- In the end, python code is python code, and it will work however you install python.

Downloading Lab Materials

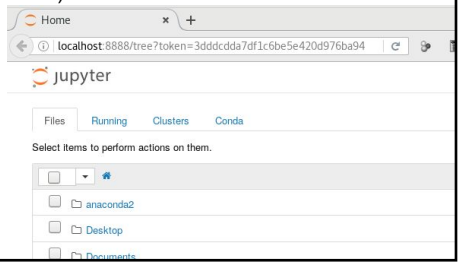
- Lab materials are here: github.com/UM-CSS
- Select the lab you want from the list
- 2 ways to download:
  - **If you use Git** already, simply clone the repository
  - **If you don't use Git,** click 

  - Then download and extract the zip file

Note: Don't double-click the ".ipynb" file. While tempting, this will not open with Jupyter or Anaconda. You must open Jupyter Notebook first, then find the file in there.

How to Use Jupyter Notebooks (Windows, Conda)

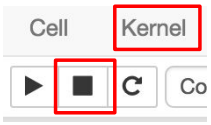
1. Search for programs with the windows or O icon. 
2. Type "jupyter" and wait for "Jupyter Notebook" to show up. (This can be slow if you just installed it.)
3. Click "Jupyter Notebook" and wait for the browser window to pop up.
4. (Ignore the terminal.)



Using Jupyter Notebooks (cont.)

- Links in Notebook browser are places on your computer.
  - If you can, use them to navigate to where you downloaded the workshop files.
  - Otherwise, click "upload" in the top right to add the files you downloaded,
  - Or click "new" to create a new notebook (".ipynb" file) to work in.

Jupyter Notebook Essentials



- If the code gets out of control, hit the “Stop” and “Kernel -> Restart” buttons at the top.
- To run the code in a cell (text box), click inside it and press “shift + enter”.
- The “Insert” menu lets you insert cells above or below your current position.
- The “Cell” menu lets you change cell types from code to headings or text (and back) to describe the code around them.
- There are hotkeys for everything: www.cheatography.com/weidadeyue/cheat-sheets/jupyter-notebook

How to Use Jupyter Notebooks (Other)

Linux, using terminal

- Navigate to the directory you want to work in
- Run `` `jupyter notebook` ``,
- Wait for browser tab to pop up

Mac

- Open “Anaconda Navigator” program
- Click “Jupyter Notebook” icon
- Wait for browser tab to pop up. (Ignore the terminal window. The terminal is the back end, but it is not interactive. Your work will be done in the browser.)

Some Reasons Jupyter Notebooks Are Great

- They’re great for exploring data. How many rows are there? What does it look like in a scatter plot? What’s the mean value? Ask questions of your data and get answers interactively.
- They’re great for testing code (and learning python). Does this line do what I want? Let’s change it and re-run just this one line of code.
- Finished programs can be exported to “.py” scripts (like “.do” files in STATA)
- Jupyter is integrated with python documentation, so you can get information about functions and tab-complete your code while you work.
- Notebooks combine the interactive, one line at a time flexibility of the terminal with the graphical interface perks of an IDE.