PPHA 30531: Data Skills for Public Policy

Jeff Levy
levyjeff@uchicago.edu
Keller 3101

Fall Quarter, 2019

Course Information

Section 1: T & Th 11:00 AM - 12:20 PM
Keller 1022

Section 2: T & Th 2:00 PM - 3:20 PM
Keller 1022

Labs: M & W 3:30-4:50 and 5:00-6:20
Keller 0023

October 1st - December 5th, 2019

Office Hours

I will be available at my office (Keller 3101) on Tuesdays and Thursdays from 10:00-11:00, or other times by appointment. TAs will be available in the lab sections listed above, or by appointment.

Teaching Assistants

Eric Langowski - langowski@uchicago.edu
Peter Li - jizhao@uchicago.edu
Ruixi Li - rxli@uchicago.edu
Krista Chan - kristachan@uchicago.edu
Joseph Nardi - nardij@uchicago.edu

Prerequisites

The course PPHA 30550, Intro to Programming for Public Policy (renamed Programming and Data Skills I going forward), is required to take this course.

I may approve exceptions, but only upon demonstration of coding skills with Python and Pandas that adequately demonstrates the skills taught in PPHA 30550.

Course Objectives

This course will build directly on the material covered in PPHA 30550. We will assume a grasp of the Python skills from the previous class at the start, so that we can focus on practical applications to research. Whereas the goal of the first class was to prepare students for entry-level policy research positions, the goal of this course will be to broaden that experience so that students are familiar with many specific situations they may encounter while working in policy research. The quarter will contiain the following elements:

- Review of projects from previous quarter
- Review of code organization using functions
- Practice dealing with the sort of datasets frequently encountered in research
- Practice reading and working with code written by others
- More practical applications of Python to research, including:
  - Irregularly shaped or unstructred data
  - Scraping data from PDF documents
  - Natural Language Processing

Additionally, we will have guest speakers throughout the quarter depending on their schedules. Confirmed so far:

- **Urban Institute**
- **Civis Analytics**
- **Chicago Coalition for the Homeless** - Julie Dworkin
- **UChicago Crime Lab** - Emma Nechamkin and Alex Williamson
- **General Dynamics Land Systems / RAND Corporation / Department of Defense** - David Bak

Speakers are subject to change, and will require a flexible course schedule on a week-by-week basis.

Software and Resources

It is strongly recommended that you bring a laptop to every class. You can obviously come without it, but it will be difficult to keep up if you cannot try things along with the lecture and during exercises.

There are no required text books for this class. As one of the most popular and fastest-growing computer languages in the world, Python is extremely well supported online. I expect students will primarily be using the official Python documentation and StackOverflow, which will be discussed in class.

However, I do suggest purchasing the text Python for Data Analysis 2nd Edition by Wes Mckinney, which is available online or in the school bookstore. Not only is it very useful both as a quick reference and when read comphrehensively as a guide, it is also written by the author of Pandas, the package used for data analysis in Python. The package is free and open-source, so this is also a good way of giving back to the creator. If you purchase this it is very important you get the 2nd edition, as the original is outdated.

If you are using the same computer that you used in PPHA 30550, then you should not have to change any software. Otherwise, there are two pieces of software that are required for this class, and three that I suggest, all of which are free:

- *You must come to class on day one* with the Anaconda Python Distribution installed already. Please select version 3.7. You can also use version 3.6 if that is what you already have installed previously. No version of Python 2.x is acceptable.

- *You must come to class on day one* with the GitHub Desktop installed. You may also use the Git command line interface if you have it installed from before and know how to use it, though it also comes as part of the Desktop download.

- The Atom open-source text editor. I suggest Atom, but perfectly viable alternatives include Sublime, Vim, and many others. You may also chose to use one of the IDEs (integrated development environments) that comes with Anaconda, such as Spyder. Whatever choice you make, please have it installed and ready to go on day one as well.

- The Notepad++ text editor. This is strictly optional, but I find this lightweight text editor useful for viewing raw data alongside my code when necessary. You can choose it as your text editor for coding, as well.

- A command line shell that offers more features than the default. This is also strictly optional, as all computers will have their own basic version. I like, for example, ConEmu for Windows, which provides handy tabs and other features that make working in it easier.

Note that overall, the software environment you choose for developing code is entirely personal preference. You simply must have some distribution of the programming language you will work in, some place to write your code, and some place to run your code. I will frequently be using Jupyter Notebooks in my lectures, which is included with the Anaconda Python distribution. This is not the way I suggest you develop your own projects (discussed further in class), but it will be the easiest way for you to follow the class notebooks.

Attendance

Attendance to a minimum of one lab per week is mandatory, and graded. There are four time slots for labs, and anyone from either section can attend any of the four. You may also, of course, attend more than one if you choose. Labs in week 1 will be strictly optional and ungraded, and will be intended for setting up software or asking specific questions about projects from last quarter.

There is no attendance policy for lecture, and you don't need to give me excuses not to come. However, you will be fully responsible for the material covered in class, and while code from class may be posted, it will rarely if ever come with the explanations provided in class.

If you experience issues with attending class or completing work due to child care, please speak with me directly so we can find an accomodation.

Academic Integrity

Standards of academic conduct are set forth in the University's Academic Integrity guide.

As it pertains to comptuer code, it is *required* that all sources are cited in the comments of any code you submit for this class. If you find a solution on StackOverflow (or anywhere else online) then you must include a comment with the relevant URL. If you work with a classmate on an assignment *you must list each other's names in comments at the top.*

Your code must always be original and uncopied, but some similarities are to be expected. This can be somewhat subjective; for example, if you find a solution that involves a list comprehension on StackOverflow you must cite that link, but if you are familiar with list comprehensions and write similar code on your own, it does not need to be cited. How this is determined will be based on your entire body of work and covered class material, and will be at the discretion of the TAs. When working with classmates, it is reasonable to discuss approaches and solutions, but I advise you to never actually look at each other's code. This way you will more easily avoid any semblance of copying.

It is expected that your assignments will be a combination of your completely original code and code you have writen based on inspiration found elsewhere. It is hard to explicitly state the appropriate balance. Note however that if "too much" of your code is determined to be unoriginal based on citations, you may be given a chance to redo it with no penalty. If "too much" of your code is determined to be unoriginal and uncited, you may fail the assignment and be subject to further disciplinary actions.

Homework, Exams, and Grading

There are no exams or final projects for this class. Your grade will consist of weekly assignments and lab attendance. Assignments will be due by the start of the first class each week over GitHub Classrooms, as judged by the timestamp on the submission versus the start of class time. Late work is not accepted without prior approval or with a documented emergency.

Your grade will be calculated as 90% weekly assignments, 10% weekly lab attendance. A minimum of 60% is required to pass this course. Among those who pass, final grades will use the standard Harris curve of 1/8 A, 1/4 A-, 1/4 B+, 1/4 B, 1/8 B-. Note that while this curve makes it hard to get a bad grade, it also makes it very difficult to earn a top grade; if there were 100 students then only about 12 would earn an A.

Course Outline

The specifics of each week will remain fluid in order to accomodate the guest speakers, but will be discussed in advance.

Week 1

- October 1st - Introduction, syllabus
- October 3rd - Review of projects

Week 2

- October 8th - Projects, review of functions and organization
- October 10th - Projects, review of functions and organization

Week 3

- October 15th - Projects, irregularly shaped data
- October 17th - Projects, irregularly shaped data

Week 4

- October 22nd - **Urban Institute speaker**
- October 24th - Working with data from the Urban Institute

Week 5

- October 29th - Natural Language Processing
- October 31st - Natural Language Processing

Week 6

- November 5th - Emma Nechamkin and Alex Williamson, **UChicago Crime Lab**
- November 7th - Julie Dworkin, **Chicago Coalition for the Homeless**

Week 7

- November 12th - Working with data from week 6 speakers
- November 14th - Working with data from week 6 speakers

Week 8

- November 19th - **Civis Analytics speaker**
- November 21st - Working with data from Civis

Week 9

- November 26th - David Bak, **General Dynamics Land Systems / RAND Corporation / Department of Defense**
- November 28th - No class - Thanksgiving break

Week 10

- December 3rd - Scraping data from PDF documents
- December 5th - Dealing with other data formats - S3/AWS, SQL, fixed width format, etc.