Introduction to Python for Social Science Course Syllabus

Musashi Harukawa

HT 2021

Course Description

Introduction to Python for Social Science is an 8-week optional methods module aimed at social science researchers seeking to learn programming skills for their research. There will be weekly lectures, lasting 60 to 90 minutes, followed by a workshop, and supplemented by weekly office hours. All of the above will be conducted on Teams and communicated on Canvas.

This course begins with an extremely brief introduction to the Python programming language, then teaches four key skills for social science research:

- Data Cleaning and Merging with `pandas`
- Data Visualization with `matplotlib` and `seaborn`
- Introductory Machine Learning with `scikit-learn`
- Automated Data Collection with `beautifulsoup` and `selenium`

As an optional course, there will be no marked assignments, but there are weekly problem sets and readings designed to aid learning. Students are encouraged to complete these tasks, and will have the opportunity to discuss them during the workshop or office hours.

This course is aimed at complete beginners, although experience with other programming languages (such as `R`) may provide some useful reference points. Note that those intending to attend the Trinity Term text-as-data module are **strongly** encouraged to attend this course, as the skills taught here will be assumed knowledge.

Information Regarding Attendance

Please read this information prior to attending, as I will not have time to help with problems that arise during lectures or the workshop.

Students will require the means to read, write, and execute code in Jupyter notebooks. For the purposes of the course, `Google Colab` is sufficient, but students who desire to have a local installation should read the attached installation guide.

Readings

The course does not follow any particular textbook. However, readings will be assigned primarily from the following books:

- Week 1: *Automate the Boring Stuff with Python* by Al Sweigart. Excellent introduction to Python and what it can do for you. Free from website.
- Weeks 2 and 3: *Python for Data Analysis: Data Wrangling with Pandas, NumPy and IPython, 2nd edition* by Wes McKinney. Available for free via SOLO.
- Weeks 5 and 6: *Elements of Statistical Learning, 2nd Edition* by Hastie, Tibshirani and Friedman. This book provides an excellent and comprehensive introduction to machine learning. Free from website.
- Weeks 7 and 8: *Web Scraping with Python, 2nd Edition* by Ryan Mitchell, 2018. Can be accessed for free via SOLO.

For further questions regarding course specifics or accessibility requirements, please feel free to write to me at musashi.harukawa@merton.ox.ac.uk.

Course Outline

Please note that do to my own evolving situation, this syllabus may be subject to change.

Week 1: Introduction to Python and the Development Environment

Learning Aims:

1. What is Python and what can I use it for?
2. What are the tools I can use to write Python code?
3. Writing your first Python script

There are three learning goals in the first week. The first relates to what Python is, and how it can be useful for social science researchers. Students will learn about the various use cases for Python, and come up with ways that it may help them achieve their research aims.

The second learning goal is to gain familiarity with the tools used to code in Python and present their research. These include Jupyter notebooks, IDEs and the terminal. Students will primarily use Jupyter notebooks in this course, but are welcome to use alternative development tools.

The final goal is to write their first program in Python. Commands and operators such as `print`, `+`, `&` etc. will be introduced.

Week 2: Data Structures and Pandas I

Learning Aims:

1. Recap: Data Structures in Base Python
2. Data I/O with `pandas`
3. Selecting, filtering and indexing data in `pandas`
4. Summary statistics in `pandas`
5. `NumPy` Data Types

The second week recaps some basic data structures in base Python from the previous week and then introduces a key library for data analysis: `pandas`. In base Python, students will learn about lists and dictionaries.

In `pandas`, students will learn how to read in various data formats, clean and index data, and produce summary statistics. Students will also be introduced to data types.

The goal of this week is to be able to use `pandas` to open `csv`, `html`, `xls`, or `dta` files, to slice and filter them, and then to produce summary statistics.

Week 3: Data Structures and Pandas II

Learning Aims:

1. Writing Python functions
2. Vectorize with `apply`
3. Split-apply-combine with `groupby`
4. Working with datetime data

The third week builds on students' knowledge of `pandas`, introducing two key tools in data analysis: `apply` and `groupby`. Students will also learn how to write functions and be introduced to the idea of namespaces.

By the end of this week, students should have a sufficient grounding in handling tabular data with base Python and `pandas` to deal with most data cleaning and reshaping tasks they use in their own research.

Week 4: Data Visualization

Learning Aims:

1. The "philosophy" of `matplotlib`
2. Figures, subplots, axes, legends
3. Plotting uni- and bivariate datasets in `matplotlib`

4. The convenience of `seaborn`
5. Customizing your plots

The fourth week introduces two key libraries for data visualization: `matplotlib` and `seaborn`. Students will learn the philosophy behind data visualization, and how to create a number of useful 2D graphs.

***Week 5*: Machine Learning I**

1. Introduction to Machine Learning
2. Introduction to `scikit-learn`
3. kmeans clustering with `scikit-learn`
4. Presenting your kmeans clustering results

The fifth week introduces to students to machine learning with the `scikit-learn` library. After discussing the aims and developments within the field, students learn about unsupervised clustering with the k-means algorithm.

***Week 6*: Machine Learning II**

1. Random Forest regression and classification with `scikit-learn`
2. k-fold cross validation with `scikit-learn`
3. Hyperparameter Tuning with `Grid` and `RandomizedSearchCV`
4. When not to use linear models

The sixth week introduces supervised machine learning with the random forest algorithm. Students then learn about cross-validation techniques and their implementation in `scikit-learn`. Finally students learn about hyperparameters, and how to choose the optimal initialising parameters for the model. The lesson ends with a discussion about the difference between prediction and explanation.

***Week 7*: Web Scraping I**

1. The Structure of Websites: `html`
2. Requesting webpages with `requests` or `urllib`
3. Parsing `html` with `beautifulsoup`
4. Introduction to regular expressions (`regex`)

Students will learn the fundamentals of writing a script to automate web-based data collection. This will include a discussion of the legality and ethics of the method, when and how it should be employed, and the potential consequences of inappropriately applying it.

Students will learn the basics of the structure of every webpage; `html`, and how the library `beautifulsoup` can help them parse and navigate this in order to extract data from webpages. Students will also learn a basic introduction to regular expressions with the `re` library.

***Week 8*: Web Scraping II**

1. Working with APIs
2. Browser automation with `Selenium`

Students will learn further techniques in web scraping, using APIs and browser automation to interact with a further variety of data sources on the web.