# Jupyter Notebooks for Python, Data Science and Econometrics

Guido Kuersteiner and Ko Miura

August 24, 2024

We provide templates for teaching materials using Python and data science tools aimed at undergraduate instruction. The notebooks are written with a typical undergraduate program in economics in mind and targeted at the intermediate to advanced undergraduate course level. We start out by explaining basic elements of the Python programming language. We then show how Python and specialized Python packages can be used to tackle simple data storage, retrieval, manipulation and analysis operations. We move on from there to explore more elaborate data analysis tasks by working through various examples drawn broadly from economics related topics. We show how standard tasks in econometrics can be handled in Python.

Python is a powerful and versatile high level object oriented programming language. In addition to the core syntax there are now over 10,000 Python packages providing additional functionality in a vast number of areas. Noteworthy capabilities of Python are in Artificial Intelligence, Machine Learning, Data Science and High Performance Computing (HPC). The capabilities of Python and its associated packages far exceed what can be covered in these notebooks, but the hope is that the interested reader will be able to explore additional topics with the help of what is discussed here. The tutorial assumes no prior knowledge of Python but some familiarity with basic coding techniques will be helpful in working through the examples. The material focuses on how things can be done in Python, and not on basic general algorithmic principles of programming with computer languages. Similarly, we assume that the reader is familiar with basic statistical and econometric techniques typically covered in intermediate and advanced undergraduate courses in economics. If used in conjunction with such courses, the instructor is expected to provide the necessary background in separate units.

There are a number of open source implementations of Python. We recommend the Anaconda implementation which can be downloaded free of charge here (https://www.anaconda.com/). The Anaconda navigator is a gateway to applications such as Jupyter and Spyder. Spyder is a graphical Integrated Development Environment (IDE) with a similar look and feel as the Matlab environment. Spyder has a code editor and debugging environment that can be used for large application development. It also has a command line system that allows to execute single Python commands, similar to Matlab. For the notebooks we use Jupyter which is a web-based programming environment that allows for interactive text and executable code. One particularly nice feature is that students can easily manipulate sample code and observe how this changes program output. This feature can be used in live class-room demonstrations to illustrate how code works. The notebooks can be edited and expanded with additional material an instructor finds useful.

To use the notebooks we recommend the following steps. Start by installing Anaconda on your computer. Anaconda is available for Windows, Mac and Linux operating systems and free of charge for individual users. Once installed, start the Anaconda navigator application. It will show all installed applications on the home screen. To use the notebooks launch the Jupyter application. On the right side of the navigator, below the 'Home' tab is the 'Enviornments' tab. It can be used to install missing packages that may be required by our notebooks. An example of a package that typically is not installed by default is 'geopandas' which is needed to represent data with a geographic orientation in maps. Section 5.3 of our notebooks covers this material and requires that 'geopandas' is installed. Before you can use the notebooks you should download them from github onto your computer. Once downloaded, start Jupyter and navigate to the location where you saved the notebook files and sample data that we provide.

While these notebooks and materials may also be helpful to researchers looking for an introduction to Python, our target is an undergraduate audience. Research level applications often require using the latest econometric techniques which we do not cover. For those looking for ready made software implementing such techniques, STATA may be the best option. Depending on the details of what exactly needs to be done it may or may not make sense to integrate STATA into Python code. We provide some basic examples of how this can be done using Stata 17 and higher.

The notebooks are divided into five modules. Unless students are already familiar with Python coding we recommend that all students be exposed to at least some of the more basic material in Part I. For a basic entry level course Notebooks 2 and 3 where we discuss user defined functions and data types as well as object oriented programming tools may be skipped. Much of Part II should be relevant for a simple data analysis course that focuses on visualization and descriptive data analysis. We also cover data storage and manipulation methods supported through Python packages. Part III uses knowledge from the two earlier parts to explore simple econometric problems, mostly using linear regression. We also cover hypothesis testing and ways to improve the flexibility of regression models using non-linear functional forms. Part IV is looking at more advanced econometric methods and shows how the functionality of Stata can be accessed from Python programs. This is an advanced topic that can be skipped in introductory courses. Finally, in Part V we discuss how Python can be used for automated data collected using web scraping techniques.

College Park, August 2024, Guido Kuersteiner and Ko Miura

## Table of Contents

### Part I: Python Programming Basics

### Part II: Data Manipulation and Data Science

### Part III: Econometrics

**Part IV: Stata and Python Integration**

**Part V: Data Collection with Web Scraping**