Python for Economics

Zhenhua He
Afternoon session, 9/17/2021
Table of Contents

This course is divided into numbered lessons

13. Matplotlib
14. Pandas
  • Wrap up
Python Libraries Covered

- **matplotlib**: Plotting data
- **pandas**: Analyzing, cleaning, and manipulating data
Lesson 13

Data visualization with Matplotlib

Use Python Matplotlib library for data visualization
Learning Objectives

After this lesson, you will know how to make:

- Scatter plot and Line plot
- Color map
- Contour figures
- 3D figures
  - Surface plots
  - Wire-frame plot
  - Contour plots with projections
Anatomy of a Scatter Plot

Marker
• style
• size
• color

Figure
• title
• xlabel
• ylabel
# Scatter plot - Marker symbols

<table>
<thead>
<tr>
<th>marker</th>
<th>symbol</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot; &quot;</td>
<td>⋅</td>
<td>point</td>
</tr>
<tr>
<td>&quot; , &quot;</td>
<td>⋅</td>
<td>pixel</td>
</tr>
<tr>
<td>&quot;O&quot;</td>
<td>⬤</td>
<td>circle</td>
</tr>
<tr>
<td>&quot;V&quot;</td>
<td>▼</td>
<td>triangle_down</td>
</tr>
<tr>
<td>&quot;^&quot;</td>
<td>⬤</td>
<td>triangle_up</td>
</tr>
<tr>
<td>&quot;&lt;&quot;</td>
<td>←</td>
<td>triangle_left</td>
</tr>
<tr>
<td>&quot;&gt;&quot;</td>
<td>→</td>
<td>triangle_right</td>
</tr>
<tr>
<td>&quot;1&quot;</td>
<td>⬤</td>
<td>tri_down</td>
</tr>
<tr>
<td>&quot;2&quot;</td>
<td>⬤</td>
<td>tri_up</td>
</tr>
<tr>
<td>&quot;3&quot;</td>
<td>←</td>
<td>tri_left</td>
</tr>
<tr>
<td>&quot;4&quot;</td>
<td>⬤</td>
<td>tri_right</td>
</tr>
<tr>
<td>&quot;8&quot;</td>
<td>⬤</td>
<td>octagon</td>
</tr>
<tr>
<td>&quot;s&quot;</td>
<td>■</td>
<td>square</td>
</tr>
<tr>
<td>&quot;p&quot;</td>
<td>⬤</td>
<td>pentagon</td>
</tr>
<tr>
<td>&quot;p&quot;</td>
<td>⬤</td>
<td>plus (filled)</td>
</tr>
<tr>
<td>&quot;*&quot;</td>
<td>⬤</td>
<td>star</td>
</tr>
</tbody>
</table>
Hot Tip!

Give a module a nickname with `as`

```python
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
```
Examples and Exercises

Go to Google Classroom assignment “Scatter Plot”

Tasks

• Follow instructions for the examples
• Work on the exercises (due by 9/17 6:00 PM)
Line plot

Simple line styles can be defined using the strings "solid", "dotted", "dashed" or "dashdot".

```
Named linestyles

solid
'solid'

dotted
'dotted'

dashed
'dashed'

dashdot
'dashdot'

Parametrized linestyles

loosely dotted
(0, (1, 10))

dotted
(0, (1, 1))

densely dotted
(0, (1, 1))

loosely dashed
(0, (5, 10))
```
Examples and Exercises

Go to Google Classroom assignment “Line Plot”

Tasks

• Follow instructions for the examples
• Work on the exercises (due by 9/17 6:00 PM)
Subplots

Vertically stacked subplots

Horizontally stacked subplots
Exercises and Homework

Go to Google Classroom assignment “Subplots”

Tasks

- Follow instructions for the examples
- Work on the exercises (due by 9/17 6:00 PM)
- Work on the homework (due by 9/23 11:59 PM)
Color map + savefig()

color map
- pcolor
- imshow

savefig()
- save the current figure
Examples and Exercises

Go to Google Classroom assignment “Color Plot”

Tasks

• Follow instructions for the examples
• Work on the exercises (due by 9/17 6:00 PM)
Break Time Reminder Slide

10 minutes break
Lesson 14
Pandas

Use Python Pandas library to manipulate data
Learning Objectives

After this lesson, you should know how to:

• Create a DataFrame
• Drop Entries
• Index, Select, and Filter data
• Sort data
• Handle missing and duplicate data
• Input and Output
# Pandas VS NumPy

<table>
<thead>
<tr>
<th>NumPy</th>
<th>Pandas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faster mathematical operations ✅</td>
<td>Slower mathematical operations</td>
</tr>
<tr>
<td>Only supports integer index</td>
<td>Customized index ✅</td>
</tr>
<tr>
<td>must use structured arrays</td>
<td>Easily handles different data types ✅</td>
</tr>
<tr>
<td>better performance when number of rows is 50K or less</td>
<td>better performance when number of rows is 500K or more ✅</td>
</tr>
<tr>
<td>more complicated to read and write files</td>
<td>simpler to read and write more file formats ✅</td>
</tr>
</tbody>
</table>
Series

- One-dimensional labeled array
- Capable of holding any data type (integers, strings, floating point numbers, etc.)
- Example: time-series stock price data
Array refresher -> Series

- index
- values
- get a value
- get a set of values
- filtering
Examples and Exercises

Go to Google Classroom assignment “Series”

Tasks

• Follow instructions for the examples
• Work on the exercises (due by 9/17 6:00 PM )

Create a series -
  index: datetime;
  values: randomly generated stock price.
DataFrame

- Primary Pandas data structure
- A dict-like container for Series objects
- Two-dimensional size-mutable
- Heterogeneous tabular data structure

<table>
<thead>
<tr>
<th></th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>x</td>
<td>0.1</td>
<td>True</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>y</td>
<td>2.4</td>
<td>False</td>
</tr>
<tr>
<td>C</td>
<td>2</td>
<td>z</td>
<td>1.9</td>
<td>True</td>
</tr>
<tr>
<td>D</td>
<td>NA</td>
<td>w</td>
<td>8.3</td>
<td>False</td>
</tr>
<tr>
<td>E</td>
<td>9</td>
<td>a</td>
<td>6.8</td>
<td>False</td>
</tr>
</tbody>
</table>
### DataFrame Example

**House Sale Data**

<table>
<thead>
<tr>
<th>id</th>
<th>date</th>
<th>price</th>
<th>bedrooms</th>
<th>bathrooms</th>
<th>sqft_living</th>
<th>sqft_lot</th>
<th>floors</th>
</tr>
</thead>
<tbody>
<tr>
<td>7129300520</td>
<td>20141013T00</td>
<td>221900</td>
<td>3</td>
<td>1</td>
<td>1180</td>
<td>5650</td>
<td>1</td>
</tr>
<tr>
<td>6414100192</td>
<td>20141209T00</td>
<td>538000</td>
<td>3</td>
<td>2.25</td>
<td>2570</td>
<td>7242</td>
<td>2</td>
</tr>
<tr>
<td>5631500400</td>
<td>20150225T00</td>
<td>180000</td>
<td>2</td>
<td>1</td>
<td>770</td>
<td>10000</td>
<td>1</td>
</tr>
<tr>
<td>2487200875</td>
<td>20141209T00</td>
<td>604000</td>
<td>4</td>
<td>3</td>
<td>1960</td>
<td>5000</td>
<td>1</td>
</tr>
<tr>
<td>1954400510</td>
<td>20150218T00</td>
<td>510000</td>
<td>3</td>
<td>2</td>
<td>1680</td>
<td>8080</td>
<td>1</td>
</tr>
<tr>
<td>7237550310</td>
<td>20140512T00</td>
<td>1.23E+06</td>
<td>4</td>
<td>4.5</td>
<td>5420</td>
<td>101930</td>
<td>1</td>
</tr>
<tr>
<td>1321400060</td>
<td>20140627T00</td>
<td>257500</td>
<td>3</td>
<td>2.25</td>
<td>1715</td>
<td>6819</td>
<td>2</td>
</tr>
<tr>
<td>2008000270</td>
<td>20150115T00</td>
<td>291850</td>
<td>3</td>
<td>1.5</td>
<td>1060</td>
<td>9711</td>
<td>1</td>
</tr>
<tr>
<td>2414600126</td>
<td>20150415T00</td>
<td>229500</td>
<td>3</td>
<td>1</td>
<td>1780</td>
<td>7470</td>
<td>1</td>
</tr>
</tbody>
</table>
Creating a Data Frame

Ways to do so:

- from Dictionary
- from Numpy array
- Read file (read_csv, read_excel, read_stata, read_html, ...)

Texas A&M University - High Performance Research Computing - Python for Economics - Fall 2021
Dictionary

For example, you have a car and its information is as below,

- brand: Ford
- model: Mustang
- year: 1964

You can create a dictionary as below

car_dict = {
    "brand": "Ford",
    "model": "Mustang",
    "year": 1964,
}

Examples and Exercises

Go to Google Classroom assignment Pandas “DataFrame-1”

Tasks

• Follow instructions for the examples
• Work on the exercises (due by 9/17 6:00 PM)
  1. Create a nation_economics DataFrame - including columns of Country, Continent, GDP, Population, GDPPerCapita
  2. Data on the next slide
### Examples and Exercises

country_economics data

<table>
<thead>
<tr>
<th>Country</th>
<th>Continent</th>
<th>GDP (Billion dollars)</th>
<th>Population (Millions)</th>
<th>GDPPerCapita (Thousand dollars)</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>America</td>
<td>18624.5</td>
<td>332.9</td>
<td>66.7</td>
</tr>
<tr>
<td>China</td>
<td>Asia</td>
<td>11218.3</td>
<td>1444.2</td>
<td>10.7</td>
</tr>
<tr>
<td>Japan</td>
<td>Asia</td>
<td>4936.2</td>
<td>126.1</td>
<td>43.6</td>
</tr>
<tr>
<td>Germany</td>
<td>Europe</td>
<td>3477.8</td>
<td>83.9</td>
<td>49.5</td>
</tr>
<tr>
<td>India</td>
<td>Asia</td>
<td>2259.6</td>
<td>1393.4</td>
<td>2.3</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>Europe</td>
<td>2647.9</td>
<td>68.2</td>
<td>42.9</td>
</tr>
<tr>
<td>France</td>
<td>Europe</td>
<td>2465.5</td>
<td>65.4</td>
<td>44.0</td>
</tr>
<tr>
<td>Italy</td>
<td>Europe</td>
<td>1858.9</td>
<td>60.4</td>
<td>34.6</td>
</tr>
<tr>
<td>Brazil</td>
<td>America</td>
<td>1795.9</td>
<td>214.0</td>
<td>9.6</td>
</tr>
<tr>
<td>Canada</td>
<td>America</td>
<td>1529.8</td>
<td>38.1</td>
<td>48.1</td>
</tr>
</tbody>
</table>
Break Time Reminder Slide

10 minutes break
DataFrame: data retrieval

- Retrieve a column
- Retrieve multiple columns
- Retrieve a row
- Retrieve multiple rows
- Drop entries
Examples and Exercises

Go to Google Classroom assignment Pandas “DataFrame-2”

Tasks

• Follow instructions for the examples
• Work on the exercises (due by 9/17 6:00 PM)

From the nation_economics DataFrame,

1. Retrieve the **GDPPerCapita** column
2. Retrieve the **United Kingdom** row
3. Drop the **Population** column
4. Drop the **Canada** row
DataFrame: operations/manipulation

• Selecting with slicing
• Filtering
• Sorting
  • sort by index
  • sort by values
Examples and Exercises

Go to Google Classroom assignment Pandas “DataFrame-3”

Tasks

• Follow instructions for the examples
• Work on the exercises (due by 9/17 6:00 PM )

From the national_economics DataFrame
1. Select the last 5 rows
2. Select the rows with the population greater than 100M
3. Sort the DataFrame by GDPPerCapita in descending order

• Work on the homework (due by 9/23 11:59 PM)
DataFrame: input and output

- Read/Write
- Different file formats
- describe()
Capstone - Candlestick Chart

A financial chart to depict price movement.

Four data values per marker:

• High
• Low
• Open
• Close
Go to Google Classroom assignment “Matplotlib-Candlestick”

Tasks

• Follow instructions for the examples
• Work on the exercises (due by 9/17 6:00 PM )
• Work on the homework (due by 9/23 11:59 PM)
Day 2 wrap-up

almost time to go home
Practice for next week

Most important skills to master

• List loops
• Filtering with conditionals
• Pandas DataFrame structure

Slides from today are available in Google Classroom
Homework Assignments

- Lesson 9: “Lists and Strings”
- Lesson 10: “National Economic Data”
- Lesson 11: “Talking Cats”
- Lesson 12: “Array Quiz”
- Lesson 13: “Matplotlib - Subplots”
- Lesson 13: “Matplotlib - Candlestick chart”
- Lesson 14: “Pandas - DataFrame operations”

Please submit your homework assignments before 9/23 11:59 PM
Turn in your in-class exercises before 6:00 PM today
Office Hours

Please come to our office hours for assistance
● M 10 - 11 am Blocker 219B
● T 10 - 11 am (on Zoom only)
● W 2 - 4:30 pm Blocker 219B
● R 2 - 3 pm Blocker 219B

Please join our slack channel for discussion
● Workspace sweeterworkspace.slack.com
● Channel hprc-econ-fall-21 (private channel)
New HPRC Help Resource

Bring Your Own Code (BYOC) sessions

These sessions are meant to help researchers overcome general Python programming hurdles in their research projects.

In person (Rooms 218A and 217B) or via zoom
Weekly on Wednesdays from 3-4:30pm through December 15.

Contact help@hprc.tamu.edu
Pandas Cheat Sheet (continued learning)

https://pandas.pydata.org/Pandas_Cheat_Sheet.pdf