

(Intro to) Data Analysis in Python

John Serences, jserences@ucsd.edu

Feb 20th, 2019

Class 05

Reminder:

- All in-class code and lecture slides can be found on GitHub
 - [https://github.com/JohnSerences/PSYC193 IntroPython W2019](https://github.com/JohnSerences/PSYC193_IntroPython_W2019)

Course Schedule (approximate)

- Week00, January 9: What is Python?, Jupyter Environment (Google Colab), First Program, Intro to object types and methods
- Week01, January 16: More on object types, lists, for loops, list comprehensions, slicing lists
- Week02, January 23: If...elif...else statements, dictionaries
- Week03, January 30: User input, while statements, try/except statements
- Week04, February 6: NO CLASS
- Week05, February 13: Midterm, writing functions
- **Week06, February 20: Classes, object-oriented programming**
- Week07, February 27: File Input/Output, data formats for files (e.g. JSON, HDF5)
- Week08, March 6: NumPy (numerical computing), Plotting (Matplotlib)
- Week09, March 13: Pandas (data frames and Seaborn)
- Final: Room/Time TBD

In-class work

- To launch a new notebook and do basic setup
 - Log in to: <https://colab.research.google.com/notebooks/welcome.ipynb>
 - Can just google: “google colab”
 - Use your AD credentials to login
 - File menu -> “New Python 3 Notebook”
 - File menu -> “Rename” (or just type in the name field in upper left corner)
 - File name should be: Lastname_PSYC193_Class00.ipynb
 - Use exactly this convention EVERY time, only updating the 00 counter (so next class is 01, etc).
 - File menu -> “Locate in drive”
 - This will launch a new window with a file list
 - Right click, Move, navigate to our shared folder and move file
 - Now all your work will be saved in our shared folder...

In class quiz on material from last week

- Review of midterm?
- Review questions from last week...
- Any more questions?

Procedural programming (PP)

- Writing code blocks sequentially to solve a problem, and place common tasks in functions.
 - Good modularization and generally easy to follow
 - Can get unruly with big projects, especially when updating
 - This happens in part because of an embedded structure – you might have some code that calls a function in a module, which in turn might call another function in another module, etc.
 - Can make debugging or revising code a nightmare

Useful links:

<https://blog.newrelic.com/engineering/python-programming-styles/>

<https://www.codementor.io/learn-programming/comparing-programming-paradigms-procedural-programming-vs-object-ori>

Example

```
%%writefile ModifyList.py

def sum_list(my_list):
    return sum(my_list)

def min_list(my_list):
    return min(my_list)

import ModifyList as ml

n = [1,2,3,4]
print(ml.sum_list(n))
print(ml.min_list(n))
```

Object-oriented programming (OOP)

- **Object-oriented:** Create specialized objects that are manipulated only through specific methods.
 - Break code into reusable functional object that are encapsulated (independent) from one another, but that together can be used to solve problems
 - Kind of like a “choose your own adventure” book (see [here](#))
 - If done thoughtfully, can get the same code reuse benefits you get from using functions (and modules)

Example

```
# define a new object type
class ModifyList:
    """class to modify lists"""
    def __init__(self, num_list):
        # initialize with list
        self.num_list = num_list

    # add two methods - one for adding,
    # one for finding the min
    def sum_list(self):
        self.sum_list = sum(self.num_list)

    def min_list(self):
        self.min_list = min(self.num_list)

# list of numbers
n = [1,2,3,4]
# create a new object, init with list 'n'
my_sum = ModifyList(n)
# call the methods
my_sum.sum_list()
my_sum.min_list()

print(my_sum.sum_list)
print(my_sum.min_list)
```

A few useful links

- [Newrelic](#)
- [Codementor](#)

In class exercises

- Please finalize by Friday at midnight so that I can grade in time to give feedback.
- We'll reconvene today at 11:30 (or so) to go over answers.
- Does anyone want to go over the exam more? If so, please let me know.