CSE 308- Lab #1
Python-Data Manipulation and Visualization

Due: Tuesday February 18th 2020, 8:00 am

Objectives:

After the lab, you should know how to

- Define a data structure in Python (e.g. numpy arrays)
- Write Python code to import tabular data (.csv) into pandas dataframes
- Write Python code to describe, manipulate, and plot pandas dataframes
- Write Python code to select data from pandas dataframe using indexing based on locations and values
- Write Python code to plot data from lists using the matplotlib package
- Write Python code to customize your plots (e.g. titles, axes labels, colors)

Introduction:

In order to make a visualization, we need data and we usually need it in an organized tablular form suitable for plotting. The pandas library provides easy-to-use data structures and data analysis tools that you can use to make your data easier to plot. An important data structure of the pandas library is a fast and efficient object for data manipulation called a `DataFrame`.

The matplotlib library is a powerful tool capable of producing complex publication-quality figures with fine layout control in two and three dimensions. While it is an older library, so many libraries are built on top of it and use its syntax

1. Import the proper libraries: Pandas and NumPy and create aliases pd, np respectively.
2. Load sample data (***car_loan.csv***) into data frame: df
3. Export Pandas DataFrames to csv. Save file name as ***out.csv***. hint: help(df.to_csv)
4. Run the command: df.info (). What do you see, how many columns? also what about number of entries for each column
5. It is often the case where you change your column names or remove unnecessary columns**.**
    a. Change the following columns names:
    Starting Balance: starting_balance
    Interest Paid: interest_paid
    Principal Paid: principal_paid
    New Balance': new_balance
    b. Remove the two columns “term”, and “Repayment”.
6. Run the command: **interest_missing = df['interest_paid'].isna(),** what do you see ?
7. Can you fix the problem in 6 above? hint: use the function df.loc.
    property **DataFrame.loc**: Access a group of rows and columns by label(s) or a boolean array.
8. Find the total = amount of **interest paid** over the course of the loan

9. Find the sum of all values across all columns
10. Convert Pandas DataFrames to NumPy arrays
11. Import the library pyplot from matplotlib and create alias plt
12. import seaborn library (wrapper of matplotlib) and create alias: sns
13. load data out.csv
14. use the loc property to find the values of the followings: month_numbe, interest_paid, principal_paid.
    For example: **month_number = df.loc[:, 'Month'].values** will return:
    array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60])
    # The values attribute converts a column of values into a numpy array
15. Check the type of the month_number array*?*
16. Plot the interest paid vs the number of months.
17. On the same graph plot the principal paid vs the number of months
    Good tutorial: https://matplotlib.org/tutorials/introductory/pyplot.html
18. you can use `plt.style.available` to select an appropriate aesthetic styles for your figures. Run the following command: plt.style.available, you should see alist of different dtyles.
19. Re-do 16 and 17 using the "plt.style.use('classic')". What did you notice different?
20. Re-do 19 using the "plt.style.use('fivethirtyeight')". What did you notice different?
21. Re-do 19 using the "plt.style.use('seaborn')". What did you notice different?
22. Add legend to your figures. Add it to be "center right".
23. Add markers and colors. The interest_rate in Black, and principal_paid in blue
24. Setting plot titles, labels choose font size of 12
    a. Set xlabel and ylabel : x:Month, y: Dollars
    b. Set Title: **Interest and Principal Paid Each Month**

25. Saving plots to files.

Legends

The `loc` (legend location) parameter accepts strings, ints, and tuples

string	int
'best'	0
'upper right'	1
'upper left'	2
'lower left'	3
'lower right'	4
'right'	5
'center left'	6
'center right'	7
'lower center'	8
'upper center'	9
'center'	10

The parameter accepts a 2 element tuple `x, y` where (0, 0) is the of the lower-leftcorner of the legend in axes coordinates.

Change Color

The `c` parameter accepts strings.

string	color
'b'	blue
'blue'	blue
'g'	green
'green'	green
'r'	red
'red'	red
'c'	cyan
'cyan'	cyan
'm'	magenta
'magenta'	magenta
'y'	yellow
'yellow'	yellow
'k'	black
'black'	black
'w'	white
'white'	white

The parameter also accepts hex strings. For instance, green is '#008000'. Additionally you can use rgb tuples.

***References**: Python Tutorial*: https://docs.python.org/3/tutorial/index.html
https://matplotlib.org/tutorials/introductory/pyplot.html
https://pandas.pydata.org/pandas-docs/stable/getting_started/dsintro.html#dataframe
https://numpy.org/devdocs/user/quickstart.html

Deliverables:

You need to run in:

1. you python code (include comments to explain your code)
2. report file that has the answers to the questions.
3. submit on Blackboard.

Grading

This lab is worth 100 points.