Lab07: Matplotlib in Python

Prelab 7

We encourage you to work together on the Prelab. The Pre Lab is not graded but will help you prepare for your lab session. If you have any questions on the material of the Pre lab, first check the book and recitation slides. If you continue to have any doubt about it, please email your recitation TA.

This prelab explains the basic list operations and plotting by using Matplotlib. In addition, we are providing with very helpful examples in the link Useful Functions.

Here is:

1. Prelab7
2. Useful functions

Environment Setup

Go to your working directory in “data.cs.purdue.edu” and create a directory “cs177/lab07”. Refer to the first lab material if you need to remember the steps to do so. Then start the IDLE Python Interpreter.

Goals of this Lab

So far you have been exposed to some of the standard features of Python. Some of these features are string and list manipulation, decision structures, loop structures and some basics of graphics. This Lab aims to introduce you a completely new feature of Python. Here you will learn how to plot graphs given a data set. We will specifically cover how to plot line graphs, bar charts, pie charts and scatter charts. Of course, part of the Lab will require you to manipulate the given data before plotting it. Thus, you will have to apply what you have learnt so far about Python in order to solve the challenges of this Lab.

The library needed to plot graphs is matplotlib.pyplot. This library is a collection of command style functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure. For example, create a figure, create a plotting area in a figure, plot some lines in a plotting area, decorate the plot with labels and so on. It is worthy to highlight that the library is stateful, namely, it keeps track of the current figure and plotting area, and the plotting functions are directed to the current axes.

In this Lab you will use just a selected group of functions of matplotlib.pyplot to accomplish the tasks. The details about these functions are in the next section.
Matplotlib.pyplot Functions

Before starting the explanation of each function, let's agree in the notation to be used throughout the entire description of the project. At the beginning of each file the library `matplotlib.pyplot` must be imported using the following statement:

```python
import matplotlib.pyplot as pyplot
```

This will allow using any of the function in the `matplotlib.pyplot` library through the pyplot.function() statement instead of matplotlib.pyplot.function(). For example, the function plot() will be called with the statement pyplot.plot() instead of matplotlib.pyplot.function().

The two main functions to be used in this lab (as long as you import the library as it was indicated) are as follows:

1. `pyplot.plot()`: Used to create line charts. Check this reference: [http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.plot](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.plot)
2. `pyplot.bar()`: Used to create bar charts. Check this reference: [http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.bar](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.bar)

As you saw in the PreLab there are functions that allow us to have the entire control of what is being plotted. A summary of the available functions in the `matplotlib.pyplot` library is available at [http://matplotlib.org/api/pyplot_summary.html](http://matplotlib.org/api/pyplot_summary.html)

General Instructions

Download the skeleton (lab7_skeleton.py) [here](http://courses.cs.purdue.edu/). In this Lab you are required to complete 3 tasks. Each task consists of defining a function to plot a piece of data you will format in the main() function. In each of these functions you will do the following:

1. Receive the parameters from the main() function.
2. Use the indicated `matplotlib` function to plot the desired graph.
3. Make the corresponding adjustments to make the graphs looks as indicated in each task.
4. Please review the PreLab and Useful Function sections above indicated.

**Note:** The main() function will format the data as it specified and call the 3 functions you will define in this Lab.

**Deliverable:** Change the name of the file lab7_skeleton.py to plot.py. Your deliverable is the file plot.py with the definition of main(), plotLineGraph(), plotSineFunction() and plotBarChart() functions.

**Very Important:** Carefully review the Prelab and Useful Functions section (see above). They contain all the material you need to complete the Lab.

**Task 1: Line Graph using the plot() function**

The dataset of this task is the following table.
Your task consists of defining the `plotLineGraph()`. The given dataset shows the amount of resident and non-resident students enrolled at Purdue from 1995 to 2013.

In the `main()` function

1. You will define the variables `x`, `y`, and `z`.
2. Variable `x` is a list with the different years.
3. Variable `y` is a list with the amount of resident students.
4. Variable `z` is a list with the amount of non-resident students.
5. You will call the function `plotLineGraph()`

In the `plotLineGraph` function:

1. Receive the parameters passed from `main()`.
2. Use the `plot()` function in `matplotlib.pyplot` library (please remember the way you must import the library).
3. Change the parameters of the `plot()` function to achieve the desired appearance.
4. Add the corresponding legend.
5. Remember the `show()` function to display the results.

An example of what your output should look like is below:
Task 2: Sine Function

In this task you will use numpy arrays. An array is a collection of elements of the same type. A numpy array is a special type of array that allows you to apply operations to all the elements of the array at the same time. Thus, for every value of an array x you are be able to obtain another array where each element is the sine of the corresponding element in x. That is, you are able to do operations like y = sin(x), where y and x are numpy arrays.

In order to complete this task you are required to import the numpy library as specified below. Then in the definition of the main() function you can obtain the desired variables before calling plotSineFunction() as follows:

```python
import numpy as np

x = np.arange(0.0, 1.0, 0.01)
y = np.sin(4*np.pi*x)

NOTE: The desired functions is sin(4*pi*x). As you can see numpy arrays contain fields and methods that allow you to process the entire array.

In general, after importing the appropriate libraries, you will do the following steps:

In main() function:

1. Declare and assign values to the variables using numpy arrays and functions as previously showed.
2. Call the plotSineFunction() with the appropriate parameters.

In plotSineFunction() function:

1. Use the plot() function for this graph.
2. Turn on the grid as indicated below. Refer to the grid() function in matplotlib.pyplot for details.
3. Add the corresponding legend.
4. Show the results.

```python
pyplot.grid(True)
```
Task 3: Bar Charts

The following table lists the enrollment of full-time undergraduate, professional, and graduate students at Purdue University from 2012-2014.

<table>
<thead>
<tr>
<th></th>
<th>2012</th>
<th>2013</th>
<th>2014</th>
</tr>
</thead>
<tbody>
<tr>
<td>Undergraduate</td>
<td>30,147</td>
<td>29,440</td>
<td>29,255</td>
</tr>
<tr>
<td>Professional</td>
<td>946</td>
<td>941</td>
<td>947</td>
</tr>
<tr>
<td>Graduate</td>
<td>8,163</td>
<td>8,407</td>
<td>8,568</td>
</tr>
</tbody>
</table>

In previous labs, you have already learned how to access elements of lists. In this task, you need to define three list variables called u, p and g for Undergraduate, Professional and Graduate respectively. For each of these lists, the first element should be the type of student. The second, third and fourth elements are the number of students in 2012, 2013, and 2014 respectively.

```python
def main()
    
    u = ['Undergraduate', x1, x2, x3]
    p = ['Professional', x1, x2, x3]
    g = ['Graduate', x1, x2, x3]
    
After defining the lists, the main() function needs to pass them as parameters to the function called plotBarChart(). In general:
In main() function:

1. Create the lists u, p and g from the dataset as it was already specified.
2. Call the plotBarChart() function passing the set of lists as parameters.

In the plotBarChart() function:

1. Define the position of the bars per year.
2. Define the data corresponding to those positions per year.
3. Plot the bars using the bar() function in the library.
4. Set title and axis labels
5. Add the ticks to the x-axis: see xticks() function.
6. Rotate the sticks.
7. Add the legend.
8. Show the results.

An example of what your output should look like is below:

![Figure 1](image_url)

**Turnin Instructions**

Run putty and login to data.cs.purdue.edu. Turn in your lab by typing:

```
$ cd cs177
$ turnin --c cs177=COMMON --p lab07 lab07
```

To verify that you did not submit a wrong file or an empty one, run the following command:
$ turnin -v -p lab07

**Grading Criteria**

<table>
<thead>
<tr>
<th></th>
<th>Max</th>
<th>Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task 1</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>Task 2</td>
<td>35</td>
<td></td>
</tr>
<tr>
<td>Task 3</td>
<td>35</td>
<td></td>
</tr>
</tbody>
</table>

From: [http://courses.cs.purdue.edu/ - Computer Science Courses](http://courses.cs.purdue.edu/)

Permanent link: [http://courses.cs.purdue.edu/cs17700:fall15:labs:lab07](http://courses.cs.purdue.edu/cs17700:fall15:labs:lab07)

Last update: 2015/10/06 16:56