Lab07: Matplotlib in Python

Matplotlib is a python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in python scripts, the python and ipython shell (ala MATLAB®* or Mathematica®†), web application servers, and six graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, errorcharts, scatterplots, etc, with just a few lines of code. For more information, please visit http://matplotlib.org/

Prelab 07

We encourage you to work together on the Pre Lab. 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 or the course instructor.

This prelab covers some matplotlib examples.

Here is PreLab07

Environment Setup

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

Important: Check if Matplotlib is properly installed on your machine by:

```python
>>> import matplotlib.pyplot as plt
```

If the above line runs without any errors, you should be good.

If not, go to the PreLab07 and install Matplotlib on your machine.

Task 1: Line Graph using the plot() function

Make sure you finished the PreLab07 before you start this task.

You will use the following data in this task:

<table>
<thead>
<tr>
<th>Year</th>
<th>Population of China</th>
<th>Population of India</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015</td>
<td>1376048943</td>
<td>1311050527</td>
</tr>
<tr>
<td>2005</td>
<td>1305600630</td>
<td>1144326293</td>
</tr>
</tbody>
</table>
This is the population statistics for China and India from 1955 to 2015. The data is taken from: http://www.worldometers.info/world-population/china-population and http://www.worldometers.info/world-population/india-population/

Here is the steps you are expected to follow:

1. Import matplotlib to be able to use it (import matplotlib.pyplot as plt)
2. Define a variable, 'china_population', as a list of integers, using the above table
3. Define a variable, 'india_population', as a list of integers, using the above table
4. Define a variable, 'years', as a list of integers which represents the years, using the above table
5. Plot 'china_population' using plt.plot() with appropriate legend
6. Plot 'india_population' using plt.plot() with appropriate legend
7. Change the x-axis values to the 'years' using plt.xticks() function
8. Add the title 'China and India population growth' using plt.title() function
9. Call plt.ticklabel_format( style= 'plain', axis = 'y' ) to ignore scientific notation in y-axis.
10. Call plt.legend() to see the legends we used in plt.plot() command
11. Call plt.show()

Your output should look like:
Save the program as `line_plot.py` and the output in `line_plot.png` in your lab07 directory. Make sure that it runs correctly.

**Task 2: Bar graph**

Make sure you finished the PreLab07 before you start this task.

You will use the following data in this task:

<table>
<thead>
<tr>
<th>Period</th>
<th>Android</th>
<th>iOS</th>
<th>Windows Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015</td>
<td>82.8%</td>
<td>13.9%</td>
<td>2.6%</td>
</tr>
<tr>
<td>2014</td>
<td>84.8%</td>
<td>11.6%</td>
<td>2.5%</td>
</tr>
<tr>
<td>2013</td>
<td>79.8%</td>
<td>12.9%</td>
<td>3.4%</td>
</tr>
<tr>
<td>2012</td>
<td>69.3%</td>
<td>16.6%</td>
<td>3.1%</td>
</tr>
</tbody>
</table>

This is the worldwide smartphone OS market share from 2012 to 2015. The data is taken from: [http://www.idc.com/prodserv/smartphone-os-market-share.jsp](http://www.idc.com/prodserv/smartphone-os-market-share.jsp)

Here is the steps you are expected to follow:

1. Import matplotlib to be able to use it (import matplotlib.pyplot as plt)
2. Define a variable, 'android', as a list of floats, using the above table
3. Define a variable, 'ios', as a list of floats, using the above table
4. Define a variable, 'win', as a list of floats, using the above table
5. Define a variable, 'years', as a list of integers which represents the years, using the above table
6. Plot 'android' using 'plt.bar()' with appropriate legend and color
7. Plot 'ios' using 'plt.bar()' with appropriate legend and color
8. Plot 'win' using 'plt.bar()' with appropriate legend and color
9. Change the title of the plot to 'Smartphone OS market share' using plt.title()
10. Change the x-axis values to the 'years' using plt.xticks() function
11. Add a x-label which is 'year' using plt.xlabel() function
12. Add a y-label which is 'Market share in percentage (%)' using plt.ylabel() function.
13. Call plt.legend() to see the legends we used in plt.plot() command
14. Call plt.show()

Your output should look like:
Save the program as `bar_plot.py` and the output in `bar_plot.png` in your lab07 directory. Make sure that it runs correctly.

**Task 3: Scatter plot**

Make sure you finished the PreLab07 before you start this task.

In this Task, we will read two files and plot the information we read from the files to a scatter plot using matplotlib.

Download these two files unzip them and put them under cs177/lab07 folder:

1. blue.zip
2. red.zip

Both of the two files contain 200 x and y values in the following comma-separated format:

```plaintext
x1, y1
x2, y2
x3, y3
...
...
x200, y200
```

For example, here is the first 5 lines of 'blue.txt':

```
1.22872654863, -0.605606200862
-0.72684345994, 0.109548335432
-0.806570656409, -0.647547282706
0.136419329589, 0.576791797401
0.210500251684, -0.684915800583
```
Here is the steps you are expected to follow:

1. Import matplotlib to be able to use it (import matplotlib.pyplot as plt)
2. Define two variables xblue, yblue (list of floats) and populate them from blue.txt
3. Define two variables xred, yred (list of floats) and populate them from red.txt
4. Plot blue points using plt.scatter() with blue color
5. Plot red points using plt.scatter() with red color
6. Add a x-label to the plot using plt.xlabel('x')
7. Add a y-label to the plot using plt.xlabel('y')
8. Add a title to the plot, 'blue.txt and red.txt using Scatter plot', using plt.title()
9. Call plt.show()

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

![Scatter plot example](scatter_plot.png)

Save the program as `scatter_plot.py` and the output in `scatter_plot.png` in your lab07 directory. Make sure that it runs correctly.

**References**

For more information about matplotlib, please visit [http://matplotlib.org/](http://matplotlib.org/)
The plotting commands mentioned above and other commands that you may find interesting can be found at [http://matplotlib.org/api/pyplot_summary.html](http://matplotlib.org/api/pyplot_summary.html)
More examples using matplotlib can be found at http://matplotlib.org/examples/index.html

**Turnin Instructions**

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

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

**Grading Criteria**

<table>
<thead>
<tr>
<th>Task</th>
<th>Percentage (%)</th>
<th>Breakdown</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task 1</td>
<td>30%</td>
<td>Plotting the main line-plot correctly (15%) Finalizing the x-y labels, title and legends. (15%)</td>
</tr>
<tr>
<td>Task 2</td>
<td>30%</td>
<td>Plotting the main bar-chart correctly (15%) Finalizing the x-y labels, title and legends. (15%)</td>
</tr>
<tr>
<td>Task 3</td>
<td>40%</td>
<td>Reading the file correctly (20%) Plotting the points with Matplotlib with labels and legends correctly (20%)</td>
</tr>
</tbody>
</table>

From: http://courses.cs.purdue.edu/ - Computer Science Courses

Permanent link: http://courses.cs.purdue.edu/cs17700:spring16:labs:lab07

Last update: 2016/02/23 12:32