Principles of Urban Informatics
Assignment 8

Posted on: 11/03/2014
Due Date: 11/10/2014

Introduction

The purpose of this assignment is to make sure you understand elementary plotting concepts covered in class, which can be reviewed in the class notes (available at the classes system). You will use matplotlib/python to produce plots, whose documentation is available at `http://matplotlib.sourceforge.net/`. The data for the exercises of this assignment are in four files: stocks.dat (problem 1), actions-fall-2007.dat (problem 2), microprocessors.dat (problem 3) and genes.dat (problem 4). They are packed into a single zip file called: data.zip (`http://vgc.poly.edu/projects/gx5003-fall2014/week8/data/data.zip`).

Problem 1 - Principles of plotting and connected symbols plot

The file stocks.dat has the stock quote at the start of each month from January 2006 to September 2008 for Apple Inc. (AAPL) and Microsoft Corporation (MSFT). Below we present the first three lines and the last two lines of this file.

```
month,apple,microsoft
2008-09,140.91,25.16
2008-08,169.53,27.29
...
2006-02,68.49,25.92
2006-01,75.51,27.06
```

(a) Apply the principles of plotting described in class (see slides available at the classes system) and in the class notes to generate a simple connected symbol plot (see class notes) for all Apple's stock quotes in the file stocks.dat. You should submit: the code to create the plot as a python file called *problem1a.py* (that receives the input *stocks.dat* filename as a command-line parameter); a screenshot of the plot called *problem1a.png*; and a text file named *problem1a.txt* with an explanation of the plotting principles you used to make this a clear plot.

(b) Using the quote of January 2006 as a baseline, directly compare the progress of Apple's and Microsoft's stock price by generating a plot using superposition

(both curves in the same plot). You should submit: the code to create the plot as a python file called *problem1b.py* (that receives the input *stocks.dat* filename as a command-line parameter); a screenshot of the plot called *problem1b.png*; and a text file named *problem1b.txt* with the conclusions you can draw from this plot.

(c) Repeat item b, but now using juxtaposition: split the two curves (i.e. Apple's stock price relative to January 2006 and Microsoft's stock price relative to January 2006) into two different plots. You should submit: the code to create the plots as a python file called *problem1c.py* (that receives the input *stocks.dat* filename as a command-line parameter); a screenshot of the juxtaposed plots called *problem1c.png*; and a text file named *problem1c.txt* describing which technique (superpostion vs. juxtaposition) makes more sense for this data and why.

Problem 2 - Histogram and number of bins

During a Scientific Visualization Course at University of Utah we collected all the assignments of the students into a data file for analysis. The file *actions-fall-2007.dat* has all the timestamps of all the actions of all the students in all the assignments: a total of 132131 actions. The first three lines of this file are:

```
timestamp
2007-09-15 21:24:56
2007-09-15 21:25:16
...
```

Create a histogram for the distribution of these timestamps and highlight the following due dates in the histogram:

Assignment	Due Date
0	2007-09-18 12:00:00
1	2007-09-18 12:00:00
2	2007-10-04 12:00:00
3	2007-10-25 12:00:00
4	2007-11-27 12:00:00
5	2007-12-15 12:00:00
6	2007-12-11 12:00:00

You should submit: the code to create the plot as a python file called *problem2.py* (that receives the input *actions-fall-2007.dat* filename as a command-line parameter). Also submit text files with answers for the following:

(a) How did you select the bins for the histogram and why? (submit answers in file *problem2a.txt*).

(b) What hypothesis can you make about the amount of work (i.e. number of actions) for the different assignments just by looking to this histogram? (submit answers in file *problem2b.txt*).

(c) What pattern can you observe for the amount of work (i.e. number of actions) close to the deadlines? (submit answers in file *problem2c.txt*).

Problem 3 - Dot plots for labeled data

Each line of the file *microprocessors.dat* (except for the header line) has two quantitative values associated with a label. The quantitative values are "year of introduction" and "number of transistors", and the label is the name of a "microprocessor" (e.g. 286, 386, 486, Pentium 4). See the first three lines of this file:

```
Processor,Year of Introduction,Transistors
Pentium 4 processor,2000,42000000
286,1982,120000
...
```

Generate two dot plots horizontally juxtaposed for these microprocessors: one for "year of introduction" and the other for "number of transistors". For "number of transistors" dot plot use log base 10 scale. You should submit: the code to create the plots as a python file called *problem3.py* (that receives the input *microprocessors.dat* filename as a command-line parameter); a screenshot of the juxtaposed plots called *problem3.png*.

Problem 4 - Correlation, scatterplots and regression

Let A, B, C, D be four genes. A scientist measured the activity (i.e. the expression) of these genes in 100 different conditions. The results are given in file *genes.dat*. Here are the first three lines of this file:

```
A,B,C,D
0.636244,0.239430,0.745650,0.900198
0.342974,0.800676,0.375399,0.457818
...
```

(a) Generate a 4 x 4 matrix of scatter plots to understand correlations between the four genes. Visually analyze the plot and rank the genes B, C, D in descending order of correlation to A. You should submit: the code to create the plot as a python file called *problem4a.py* (that receives the input *genes.dat* filename as a command-line parameter); a screenshot of the scattermatrix plot called *problem4a.png*.

(Extra) Now draw a linear best fit line in the plots of A with its most correlated gene, a cubic best fit curve in the plots of A with its second most correlated gene and a degree-5 polynomial best fit curve in the plots of A with its most uncorrelated gene. You should submit: the code to create the plot as a python file called *problem4b.py* (that receives the input *genes.dat* filename as a command-line parameter); a screenshot of the scattermatrix plot with the requested lines called *problem4b.png*.

How to submit your assignment?

Your assignment should be submitted using the NYU Classes system. You should submit all the requested files in each problem in a zip file named NetID_assignment_8.zip, where you should change *NetID* by your NYU Net ID.

Grading

We should be able to reproduce your plots by running your scripts with the specified parameters. The grading is going to be done based on your plots and justifications for the choices you made. It is part of the assignment to understand (and research if necessary) the concepts asked. That is especially true for the extra points in this assignment. Try to test your code before submitting: your script solves the problem when the plot is created as requested.

References

- *Matplotlib reference (installation and documentation)*: `http://matplotlib.org/`
- *Elementary plotting concepts*: Check slides and notes in the classes system

Consult the Matplotlib documentation to find out how to create each plot. *Googling* for example code for the requested plots is encouraged.