PH2150 Scientific Computing Skills

Andrew Casey

October 22, 2012

This week we are going to continue learning how to manipulate and visualise arrays. These exercises focus on two of the plotting packages that are part of the enthought distribution, *matplotlib* and *mayavi*:

- *matplotlib* is probably the single most used Python package for 2D-graphics. It provides both a very quick way to visualize data from Python and publication-quality figures in many formats.

- http://matplotlib.org/ is an online resource showing many examples, clicking on a graph in the gallery opens the code that produced it.

- *mayavi* 3D Scientific Data Visualization and Plotting

- http://docs.enthought.com/mayavi/mayavi/auto/examples.html#example-gallery mayavi example gallery including code.

1 Exercise Wk4Ex1:

Creating plots with matplotlib. The task is to replicate the plots shown in figure 1. The two plots should be produced within the same figure (*in python these are referred to as subplots*).

- The x data can be generated using linspace()
- Arrows etc. can be found by using annotate
- The text can be produced using LaTex syntax.

After generating the figure save the output as a .png file (Portable Network Graphics Image)

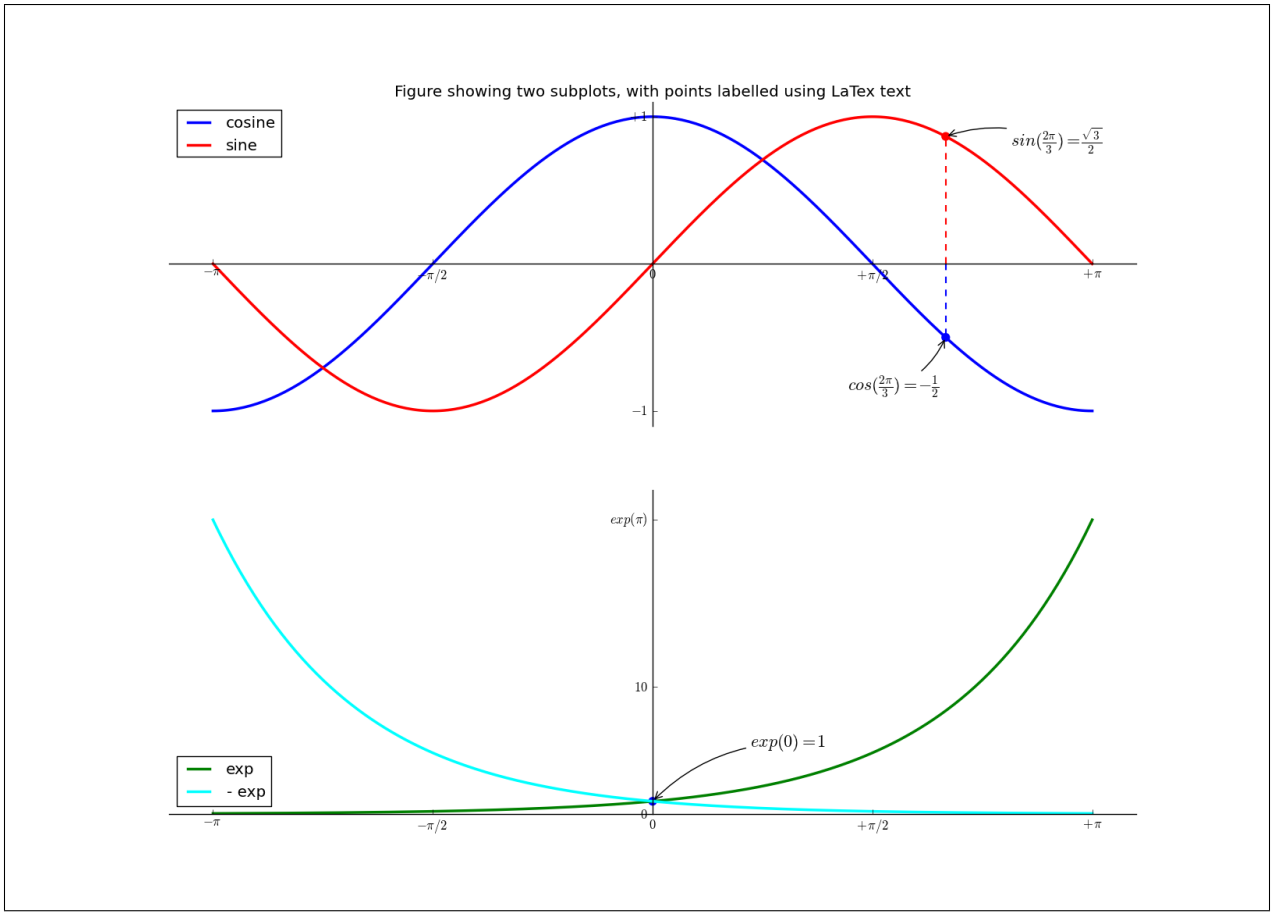

Fig. 1: A matplotlib figure configured as two subplots

2 Exercise Wk4Ex2:

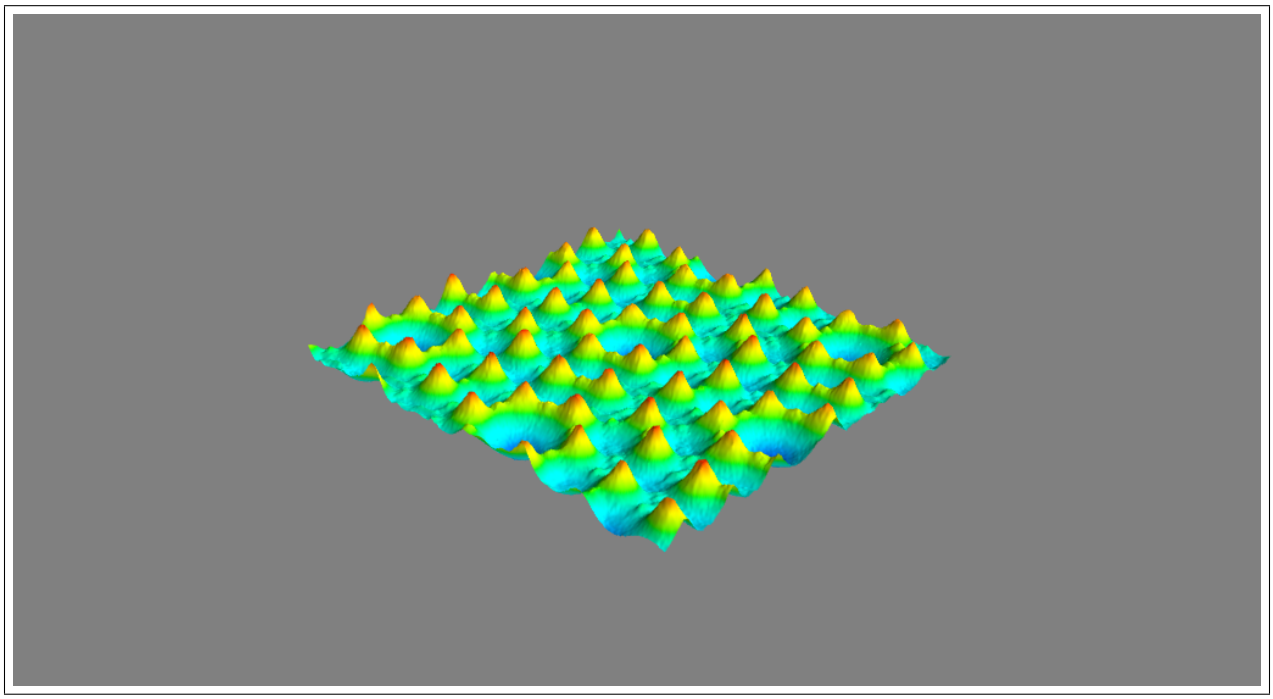

Fig. 2: Week 3 's STM data plotted using Mayavi

Mayavi is a powerful 3D visualisation tool, that comes as part of the Enthought distribution. It can be operated from within the Pylab shell (running a script) or using its own graphical interface. Using the package *mlab* from *mayavi* create a 3D visualisation of the STM data that you plotted in week 3, an example of what can be produced is shown in figure 2. After running the script from pylab, the mayavi figure window will open, within this window save an isometric view as a .png file.

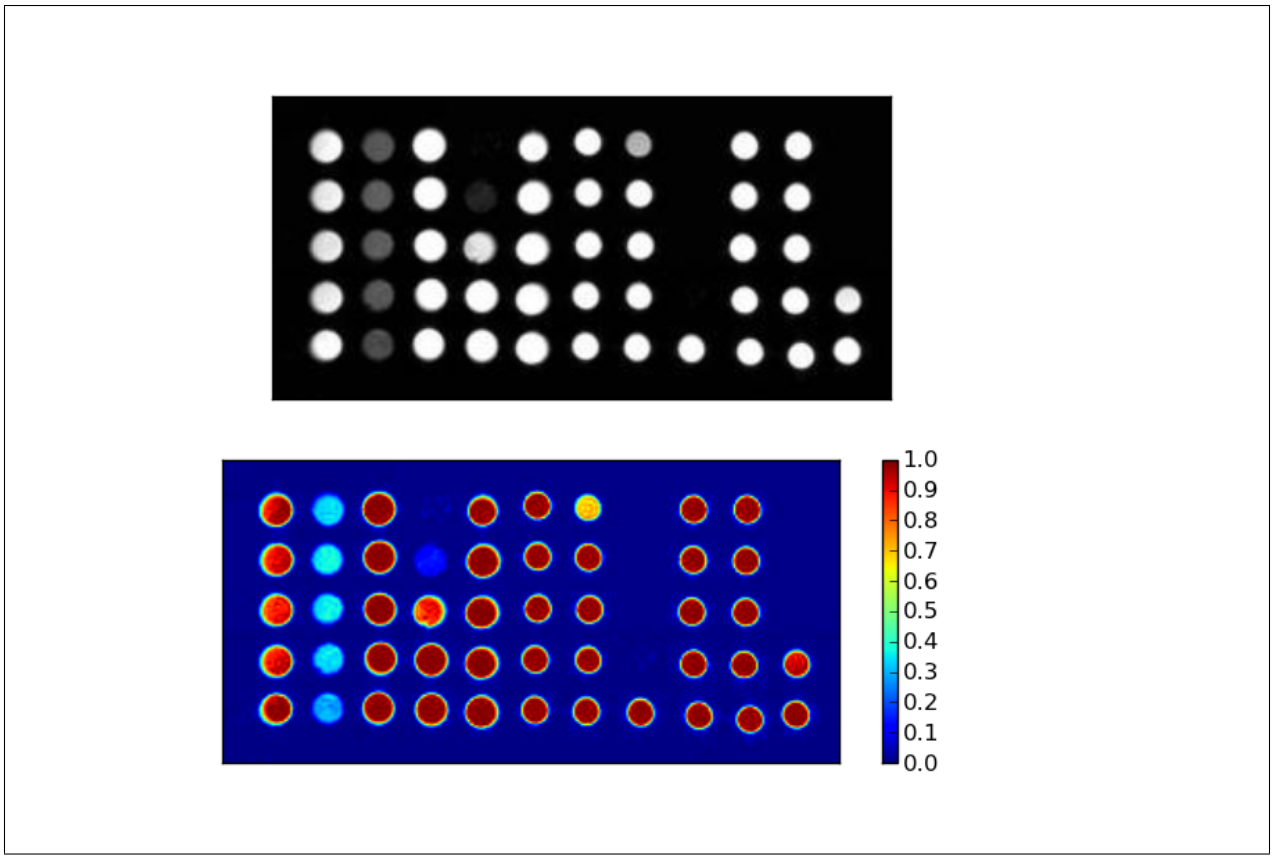

Fig. 3: Top subplot, NMR phantom loaded from 'NMR_Phantom.png', bottom subplot shows pseudo colour image obtained by slicing the RGB image to a single channel.

3 Exercise Wk4Ex3:

Importing image data into numpy arrays: import matplotlib.image as mpimg Package contains mpimg.imread, this returns an array from an image. For greyscale images the array is MxN, for RGB images the array is MxNx3. The base package can only deal with .png files, however the Python Image Library (PIL)comes as part of the enthought distribution which can deal with other image file types. In MRI the response to a contrast agent can be charaterised using NMR phantoms, arrays of liquids with different concentrations of the agent. Figure 3 shows an image of a phantom, the lower subplot shows the result of slicing the array into a single colour channel.

- Import the image 'NMR_Phantom.png' as an array
- Convert the image to single channel and show as a luminosity image, add a colour bar with scale.
- *BONUS MARKS: Use edge detection (or other method) to identify the phantoms and measure their mean intensity.*