PH2150 Scientific Computing Skills

Andrew Casey

June 2, 2013

This week we are introducing arrays as a Python construct. We have seen that by using the *%timeit* statement we can compare how long Python takes to add 1 to a million points in a list, with adding 1 to a million points in an array. Showing it is much faster to manipulate arrays. [remember: *timeit?* returns help file , list = range(1000000), array = arange(1000000)]

Also the *NumPy* package contains *ufuncs* (Universal functions that act on arrays) which reduce computation time compared to functions acting on individual elements in a list.

1 Problem Sheet 3, Exercise 1: Creating arrays

Write code to create and print the following arrays:

a) A two dimensional floating point array with 4 rows and 4 columns all filled with 0.0, call this array PS3a.

b) A two dimensional floating point array with 4 rows and 4 columns filled with the range of numbers 1-16, call this array PS3b.

c) A two dimensional floating point array with 4 rows and 4 columns filled with random numbers, call this array PS3c.

d) Use linspace() to create a two dimensional array with 4 rows and 4 columns. The first row containing numbers between 1 and 17, the second row containing 1 and 2, the third row containing numbers between 100 and 200, the fourth row should be equal to the 3rd row rounded to to decimal places. Call this array PS3d.

The next exercises are to demonstrate how basic arithmetic works on arrays:

e) Add PS3a to PS3b to PS3c, call the answer PS3e, return the element that is 4 across on the third row.

f) Multiply PS3b by itself, return PS3f take the dot product of PS3b with itself and calculate the difference from PS3f, comment on what these two operations are doing.

g) Calculate sin(x)**2 for each element of PS3c, return as PS3g.
h) Transpose the elements of PS3c.
i) Return the diagonal() of PS3c.
j) Create a (8,4)array by stacking PS3a and PS3b.
k) Create a (4,8) array by stacking PS3a and PS3c.

2 Problem Sheet 3, Exercise 2: Plotting with arrays

The file stars.dat contains a catalog of temperatures and magnitudes for 7860 nearby stars, create an array using this data. Then plot a HertzsprungRussell diagram using this catalog.

3 Problem Sheet 3, Exercise 3: Advanced plotting with arrays

The file stm.dat contains a grid of values from a scanning tunneling microscope measurement of the [111] surface of silicon. The STM is a device that measures the shape of a surface at the atomic level by tracking a sharp tip over the surface and measuring quantum tunneling current as a function of position. The end result is a grid of values that represent the height of the surface. Use imshow() to create a density plot to represent the surface of the silicon, explore the options to create a clear image of the surface.