ASTR4004/ASTR8004
Astronomical Computing
Assignment 4

Christoph Federrath, Michael Ireland, Mark Krumholz

due Thursday, October 26, 2017

1 Python project 1 – Markov Chain Monte Carlo

In this assignment you will use `emcee` in python (http://dfm.io/emcee/current/) or on github. You will simulate a periodic data set and fit a function to it. This could be, e.g., a photometric dataset from Kepler, or a series of radial velocity points. Some skeleton code (with many gaps!) is included on the course web page.

1. Create a function using python and `numpy` that simulates data that take a periodic function with a form:

$$v = a_0 + a_1 t + a_2 \sin(a_4 t) + a_3 \cos(a_4 t) \tag{1}$$

   You should simulate data at a number of random times over an interval, and include Gaussian errors for the data. The inputs a_i should take the form of a 1-dimensional python array.

2. Setting $a_0 = 0$, $a_1 = 1$, $a_2 = 1$, $a_3 = 1$ and $a_4 = 0$, simulate a data set from times $t = 20$ to $t = 35$, containing 100 points with Gaussian errors with uncertainty 0.5.

3. Use `emcee` to fit to this dataset. Plot histograms of the fitted parameters – do the results make sense? Are any of the parameter fits correlated?

4. (advanced) Show that the following is a re-parameterisation[1] of Equation (1):

$$v = a_0 + a_1 t + a_2 \sin(a_3 t + a_4) \tag{2}$$

   Which is better – Equation (1) or Equation (2) for a reliable run of `emcee`, and why? Is there a way the equation could have been re-parameterised to remove the correlation between a_0 and a_1?

5. (extra mark) If Equation (2) is your model with uniform priors in all parameters but Equation (1) is used in `emcee` instead, this produces an implicit prior on a_2. What is it?

Include all python code in your assignment, as well as a write-up.

[1] Re-parameterisation means that the second set of a_0 through a_4 are different parameters.

2 Python project 2 – numerical solution of differential equations

Consider a simple harmonic oscillator: a spring with spring constant k is attached to a mass m, and the displacement of the mass from its rest position is x. The mass experiences a restoring force,

$$F = -kx. \tag{3}$$

At time $t = 0$, the mass is released at rest at the initial position $x(0)$, and is allowed to oscillate.

2.1 Part 1

Write a python function that takes as inputs the value of the spring constant k, the mass m, the initial displacement $x(0)$, the amount of time for which to integrate T, and the number of times N at which the position should be recorded, and returns the position and velocity of the mass at times 0, $T/(N-1)$, $2T/(N-1)$, $\ldots$, T. Verify that your code matches the analytic solution for $x(0) = 0.1\,\mathrm{m}$, $k = 50\,\mathrm{N/m}$, and $m = 1\,\mathrm{kg}$.

2.2 Part 2

Modify your routine to that it works for a nonlinear spring; one with a restoring force

$$F = -k_1 x - k_2 x^3. \tag{4}$$

Make a plot comparing the solution for a simple harmonic oscillator with the parameters given in Section 2.1 and the solution for a non-linear oscillator with the same values of $x(0)$, k_1, and m, and a nonlinear coefficient $k_2 = 10^3\,\mathrm{N/m^3}$.

Please send your scripts/responses/produced figures and write-ups for these projects to mailto: christoph. federrath@ anu. edu. au *by the assignment deadline.*