NUMERICAL AND MATHEMATICAL METHODS

June 24, 2021

EXERCISE 1. Download the file fourier.dat. This text file is organized in two columns. Column 0 shows the time in seconds. Column 1 is a noisy signal y measured as a function of time.
Write a python script that

A. reads columns 0 and 1 of this file into two numpy arrays t and y, respectively;

B. plots y as a function of t with a line plot. A linear plot will be fine;

C. calculates the Fourier coefficients with the numpy.fft.rfft function;

D. re-does the same as point C., but with a user-provided discrete Fourier transform;

E. plots the absolute values of the Fourier coefficients $|c_\mathrm{k}|$ as a function of k;

F. calculates and prints the period associated with the signal.

The plots for points B. and E. do not need to be saved to a file. Just add the command plt.show() inside the script.

Please upload the script developed to address points A., B., C., D., E. and F. as a unique script named exercise1.py.

EXERCISE 2. The non-linear equation

$$x = 2 - \exp(-x) - \sin(x^2) \tag{1}$$

admits four solutions in the interval $[-\infty, 3]$. Write a python script that addresses the following points.

A. Using the relaxation method, find at least one of these solutions.

B. Using the bisection method, try to find all the four solutions. You will have to start from different guesses for the initial interval.

Suggestion: to better visualize the problem (especially for point **B.**), plot the following function

$$f(x) = x - 2 + \exp(-x) + \sin(x^2). \tag{2}$$

Please upload the script developed to address points A. and B. as two different scripts named exercise2A.py and exercise2B.py.

EXERCISE 3. Download the file five_bodies.txt. It contains the information about the position, velocity and mass of five bodies subject to gravitational force, with acceleration:

$$\vec{a}_i = -G \sum_{j \neq i, j \leq 5} m_j \frac{\vec{x}_i - \vec{x}_j}{|\vec{x}_i - \vec{x}_j|^3}, \tag{3}$$

where i is the index of the body, $j = 1, 2, 3, 4, 5$ is an index that runs over the five bodies, G is the gravity constant, m_j is the mass of the jth body, $\vec{x}_i$ is the position vector of the ith body and $\vec{v}_i$ is the velocity vector of the ith body.

In the aforementioned file, columns 0, 1 and 2 are the x, y and z components of the initial position (in astronomical units, AU); columns 3, 4 and 5 are the x, y and z components of the initial velocity (in astronomical units per year, AU/yr); column 6 is the mass in Earth masses ($\mathrm{M_{Earth}}$). Write a python script that addresses the following points.

A. Integrate the evolution of the five bodies with a leapfrog scheme between time $t = 0$ and $t = 3$ yr, with steps of 10^{-2} yr.

B. Plot the time evolution of the x, y components.

C. Calculate and plot the total energy as a function of time.

D. Calculate and plot the magnitude of the angular momentum as a function of time.

E. Can you figure out if the five bodies resemble some astrophysical system you know?

Suggestion: Remember to set the right units for the gravity constant G.

Please upload the script developed to address points A., B., C. and D. as a unique script named exercise3.py. Reply to point E. with a comment at the beginning of exercise3.py

EXERCISE 4. Please, download the file exoplanet.eu_catalog.csv, which contains observational information about the exoplanets we currently know (the file is taken from `http://exoplanet.eu/catalog/`). Column 2 [starting from zero] shows the mass M of the planet in units of Jupiter masses (M_{Jup}) and column 8 shows the radius R of the planet in units of Jupiter radius (R_{Jup}). Note that this is a CSV file: the columns are separated by a comma rather than by a space.

Write a python script that

A. reads columns 2 and 8 of this file.

B. plots R versus M with a scatter plot (please, remember to add labels and units);

C. prints the number of planets in the catalog;

D. prints the number of planets in the catalog that have both a mass and a radius measurement.

Suggestion: with **pandas**, you can read csv files in a very easy way. Reading the file with pandas, the missing masses and radii are marked as Not a Number (NaN).
Alternatively, you can read the files with **numpy.genfromtxt()**. If you prefer this latter option, remember to add the option delimiter=",".

The plot for point B. does not need to be saved to a file. Just add the command plt.show() inside the script.

Please upload the script developed to address points A., B., C. and D. as a unique script named exercise4.py.