NUMERICAL AND MATHEMATICAL METHODS, GROUP A

January 25, 2021

EXERCISE 1. Please, download the file BBH_first_gen.txt, which contains information about 2×10^5 simulated binary black hole mergers from dynamical interactions in a nuclear star cluster. **If your internet connection is very slow and you have problems in downloading the file BBH_first_gen.txt, download instead the file BBH_first_gen_small.txt, which contains just the first 10001 lines of the above file.**

Columns 1 and 2 (starting from column 0) contain information about the mass of the primary (m_1) and that of the secondary black hole (m_2) in solar masses ($M_\odot$).

Write a python script that

A. reads columns 1 and 2 of this file (the first column is column zero) into two numpy arrays m_1 and m_2, respectively;

B. plots m_2 versus m_1 with a scatter plot;

C. plots m_2 versus m_1 with a two dimensional histogram (the colors indicate the number of systems with a given combination of m_1 and m_2);

D. calculates the mean values of m_1 and m_2 and their standard deviations.

The plots for points B. and C. 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. and D. as a unique script named exercise1.py.

EXERCISE 2. According to the observations by Sana et al. (2012), the orbital period P of massive binary stars in open clusters follows a distribution:

$$p(T)\,dT = C\,T^{-0.55}\,dT, \tag{1}$$

where $T = \log_{10}(P/day)$.

A. Using the inverse random sampling, draw 10^6 random numbers from the above distribution, assuming minimum and maximum values of T equal to $T_{\min} = 0.15$ and $T_{\max} = 5.5$, respectively.

B. Plot an histogram of the randomly generated values of T and compare it with the expected distribution $p(T)$. A log−log plot is recommended. Note that it is not requested to plot the periods P, you must just plot the values of T, i.e. the logarithm in base 10 of the periods. The plot 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. and B. as a unique script named exercise2.py.

EXERCISE 3. Consider the following system of linear equations:

$$\begin{bmatrix} 0. & 5. & 15. & 0. & 1. \\ 1. & 0. & 2. & 13. & 9. \\ 17. & 5. & 17. & 2. & 1. \\ 1. & 12. & 4. & 13. & 9. \\ 1. & 2. & 4. & 13. & 0. \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 17. \\ 27. \\ 17. \\ -1. \\ 12. \end{bmatrix} \tag{2}$$

A. Write a script that solves the above system using the numpy.linalg.solve() function, which exploits the LU decomposition with pivoting.

B. Write a script that solves the above system using the Gauss-Seidel method. Note that you need to do the pivoting. Suggestion: re-arrange the matrix and array of known values so that their lines 0,1,2,3,4 become lines 2,3,0,4,1, respectively. Otherwise the algorithm may not converge.

C. (optional): Update your Gauss-Seidel script (done for point **B.**) by adding a Monte Carlo algorithm to re-arrange the matrix and the array of known values, and iterate the pivoting with the Monte Carlo algorithm till convergence is reached.

Please upload the scripts developed to address points A., B., and C. as three separate scripts, naming them exercise3_A.py, exercise3_B.py and exercise3_C.py.

EXERCISE 4. Consider a system of two point masses $i = 1, 2$ subject to Newton gravity acceleration:

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

Assuming $G = 1$, the masses of the particles are $m_1 = 3$ and $m_2 = 1$, the initial positions are and $\vec{x}_1 = (0, 0)$ and $\vec{x}_2 = (1, 0)$ and the initial velocities are $\vec{v}_1 = (0, 0)$ and $\vec{v}_2 = (0, 2)$.

A. Write a script to integrate the above system between time $t = 0$ and time $t = 5$ with the Euler method. Suggestion: use a fixed timestep $h = 0.001$.

B. Plot the time evolution of the positions of the two particles in the x, y plane, choosing the center of mass as reference frame.

C. Calculate and plot the energy variation of the system $[E(t) - E_0]/E_0$, where $E(t)$ is the energy at time t for $t \in [0, 5]$ and $E_0 = E(0)$ is the initial energy, and plot it as a function of time.

The plots 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., and C. as a unique script named exercise4_ABC.py.

D. Still using the Euler method, write a new script to integrate a system identical to the previous one, but for which the acceleration is given by the Newton equation plus a drag term:

$$\vec{a}_i = -G \sum_{j \neq i} m_j \frac{\vec{x}_i - \vec{x}_j}{|\vec{x}_i - \vec{x}_j|^3} - \beta \vec{v}_i, \tag{4}$$

where $\beta = 0.1$.

Redo points **B.** and **C.** with the new script.

The plots 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 point D as a unique script named exercise4_D.py. If you have done everything correctly, exercise4_ABC.py and exercise4_D.py will differ just by a few lines.