

MA124 Maths by Computer: Assignment 2

A. The SIR model with birth and death (10 marks)

A simple SIR model including birth and death is given by the following system of ODEs

$$\dot{S} = \mu N - \frac{\beta SI}{N} - \mu S, \quad \dot{I} = \frac{\beta SI}{N} - \gamma I - \mu I, \quad \dot{R} = \gamma I - \mu R,$$

where S is the number of susceptible individuals, I is the number of infectious individuals, R is the number of recovered or removed individuals. N is the total number of individuals, i.e. $N = S + I + R$. One can easily verify that $\dot{N} = 0$ by adding the three equations. Hence N is constant. β , γ and μ are parameters: β is the average number of contacts per person per time, γ is the reciprocal of the time an individual remains infectious, μ is both the birth rate and death rate -- these rates are taken to be the same in this model. All newborns enter the susceptible category at rate μN , while deaths occur uniformly for each category. Whether or not this model is realistic, we can examine its solutions with numerical simulations. (You can find a readable treatment of this model [here](#).)

Assignment: Write Python code to solve the above SIR model and examine the solution for one set of parameter values and one initial condition.

Details:

- You should produce two plots: a time series showing $S(t)$, $R(t)$, $I(t)$ and a phase portrait in the (S, I) plane.
- You are free to choose parameters as you wish. However, you should make μ large enough to be able to see the effect of birth/death compared with the $\mu = 0$ case.
- You are free to choose initial conditions as you want, but be sensible.
- On the phase portrait, indicate the starting and ending point for the trajectory, and also the point on the trajectory where I reaches its maximum. (You will have to work out how to find this maximum point.) Print the values of all these quantities.

You are free to use the material in the SIR case study as a starting point. So by "write Python code" we mean you start from the example. Please consider other initial conditions. If you use values for β and γ for Covid-19 (there are many different values reported on the internet), you should note that in your submission. This is not required. Using a realistic birth/death rate will have a minuscule effect on the behaviour for Covid-19. Try values for μ in the region of 1% - 5% of the value of γ . Expect to vary the time range of the plot compared with the SIR case study.

Much of this is already done for you in the case study. Use it as a model. Take the time to make a nice submission. If one collapses all the code cells, one should be able to understand the figures from Markdown text.

B. The pendulum with damping (10 marks)

A model for a damped pendulum is

$$\ddot{\theta} = -\sin \theta - \mu \dot{\theta}$$

where θ is the angle of the pendulum from the downward vertical (negative y direction) and μ is a parameter governing the strength of the damping. For $\mu > 0$, all solutions (except for those with the pendulum exactly inverted), come to rest with the pendulum at $\theta = 0 \bmod 2\pi$.

Assignment: Write Python code to solve the damped pendulum equation and examine its solutions.

Details:

- Solve the second-order equation by first writing it as a first order system.
- Fix a value for μ and find 3 initial conditions that generate solutions as follows:
 - one solution should come to rest at $\theta = 0$,
 - one should come to rest at $\theta > 0$,
 - one should come to rest at $\theta < 0$.
 - In all cases the resting value should obey $-5\pi < \theta < 5\pi$.
- Produce three plots, each with a time series of $\theta(t)$ and its derivative $v = \dot{\theta}$.
- Produce a fourth plot showing a (θ, v) phase plane in which all three trajectories are plotted. Indicate starting and end points of the trajectories.

You are free to use the material in the pendulum case study as a starting point. So by "write Python code" we mean you start from the example. You should first explore values of μ and various initial conditions to get a sense of the dynamics of the equation. Plan to spend some time with this. You may have to vary the final time and plot limits significantly from the case study. You should focus on producing a nice fourth plot.

We will allow up to 5 trajectories in the fourth plot if you wish, but only 3 should be plotted as time series. If you want to show fixed points (where $\dot{\theta} = \dot{v} = 0$), you may do so.

The case study provides a lot of help. Take the time to make a nice submission. If one collapses all the code cells, one should be able to understand the figures from Markdown text.

Further 5 marks

A further 5 marks will be awarded for each assignment based on overall quality and clarity of the submission; the level of understanding demonstrated; originality, creativity and engagement.

Submission

You will submit **two Jupyter notebooks**, one for each part. These must be `.ipynb` files, not pdf files or any other file type. Your guide is the case studies. Your submission should be such that:

- If the notebook is run and all code cells are collapsed, the notebook should be readable as a **short** report, primarily consisting of figures and descriptions of the figures. Please vary the text from the case studies.
 - We still have not discussed the level of commenting expected in Python code, but use the case studies at a guide. Make sure your functions have comments or docstrings describing what they do. One assumes the reader understands Python. Add comments to set off blocks of code or to note anything tricky. In most cases Python code explains itself.
-