DOING PHYSICS WITH PYTHON

DYNAMICAL SYSTEMS [1D]

Population Dynamics 2

Exponential Growth and the Logistic equation

Ian Cooper
matlabvisualphysics@gmail.com

DOWNLOAD DIRECTORIES FOR PYTHON CODE

Google drive

GitHub

cs_001.py   cs106.py

INTRODUCTION

Population growth models describe how the size of a population changes over time. The two fundamental models are exponential growth, which shows a population growing at an ever-increasing rate with unlimited resources, and the logistic model, a more realistic model where growth slows and levels off at the environment's carrying capacity, *K* due to resource limitations.

SIMULATIONS

Exponential growth

Consider the very simple difference equation for the variable x

$$x_{t+1} = a\,x_t$$

where x_{t+1} is the value of x at time step t+1 and x_t is the x value at time step t, a is a model parameter and is constant, and the initial condition is given by the value of x_0.

The analytical solution is

$$x(t) = x_0\,e^{b\,t} \quad b = \log\left(x_n / x_0\right) / t_n$$

where the value of the constant b is found from the value of x_n at time step n. Thus, the variable x will increase exponentially with time as shown in figure 1 where the initial condition is $x_0 = 1$ and $a = 1.2$

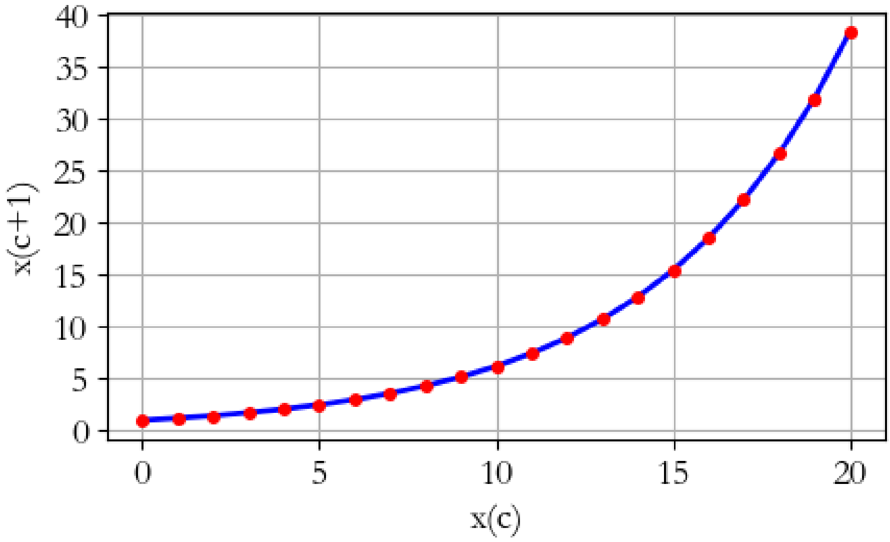

Fig. 1. Blue curve: solution of the difference equation
Red dots – exponential function
$x_0 = 1 \quad a = 1.2$ cs_001.py

Fish population growth with constant harvesting

We can model the growth of a fish population when there is a constant removal of fish (harvesting) by the simple linear difference equation

$$x_{t+1} = a\, x_t - b$$

where x is the scaled fish population with initial condition given by x_0, and a and b are positive constants. The results of the simulation are shown in figure 2.

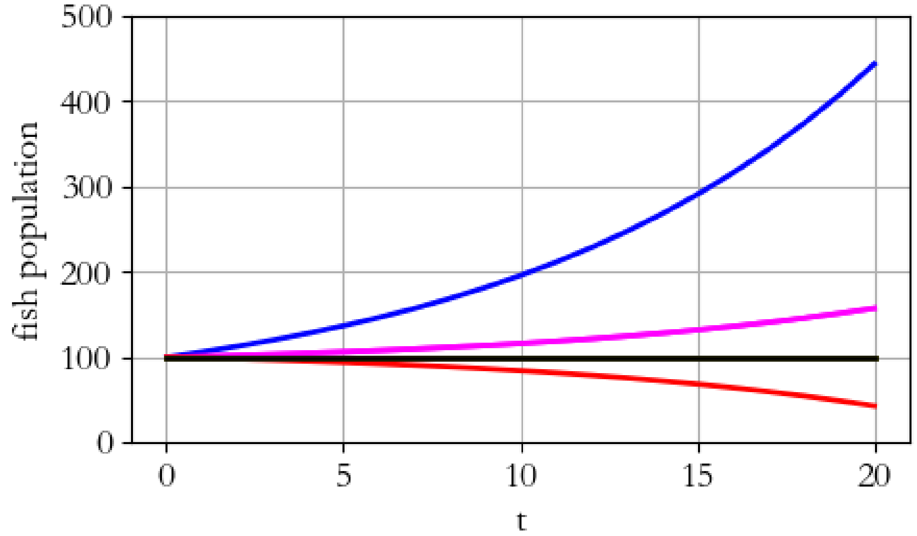

Fig. 2. Time evolution of scaled fish population: x_0 =100 and a = 1.1. **Blue curve *b* = 4**, **magenta curve *b* = 9**, **red curve *b* = 11**, and **black curve *b* = 10**.

$x_0 = 1$ $a = 1.1$ **cs_001.py**

From the plot displayed in figure 2, it is obvious why this difference equation is so useful in describing the evolution of a fish population. With $x_0 = 100$, for of $b < 10$ then the population will grow exponentially and if $b > 10$ then the population will become extinct. However, if $b = 10$, the there is a stable equilibrium and the population remains constant at 100 only if the initial population is 100 ($x_0 = 100$).

The steady-state solution x_{ss} occurs when $x_{t+1} = x_t$, so

$$x_{ss} = \frac{b}{a-1}$$

If $a = 1.1$

$b = 10$, then $x_{ss} = 100$

$b > 10$, then xss = 0

$b < 10$, then $x_{ss} \to \infty$ as shown in figure 2.

If $a \leq 1$, then $x_{ss} = 0$

Population dynamics: LOGISTIC EQUATION cs106.py

A simple mathematical model for the dynamics of a population is

$$\dot{N} = r\,N\left(1 - \frac{N}{K}\right) \qquad \textbf{logistic equation}$$

where N is the population, $\dot{N}$ is the rate of change of the population, r is the growth rate (positive constant), and K is the carrying population (equilibrium population). For the logistic equation, $\dot{N} / N$ is linearly related to the population N.

The equilibrium points (fixed points) of the system are

$$N_e = 0 \quad \text{and} \quad N_e = K$$

To check the stability of the equilibrium, we need to consider the function

$$f\,' = df \,/\, dN\big|_{N_e} = r\left(1 - (2/k)N_e\right)$$

$$N_e = 0 \quad f\,' = r > 0 \Rightarrow \quad \textbf{unstable}$$

$$N_e = K \quad f\,' = -r < 0 \Rightarrow \quad \textbf{stable}$$

For a small population the initial growth is exponential, but it slows down as the population approaches the carrying capacity K. The growth curve resembles an "S" shape. For a large population, there is an initial exponential decrease in population and then a lower decline as the population approaches the carrying capacity K (inverted "S" shape)

The population N is always positive ($N > 0$), since it makes no sense to think about a negative population and if $N(0) = 0$ then there's nobody around to start reproducing, and so the population would be zero for all time ($N(t) = 0$). $N_e = 0$ is an unstable fixed point, so a small population will initially grow exponentially away from $N \sim 0$.

$N = K$ is a stable fixed point, thus, if N is disturbed slightly from K, the disturbance will decay monotonically back to K.

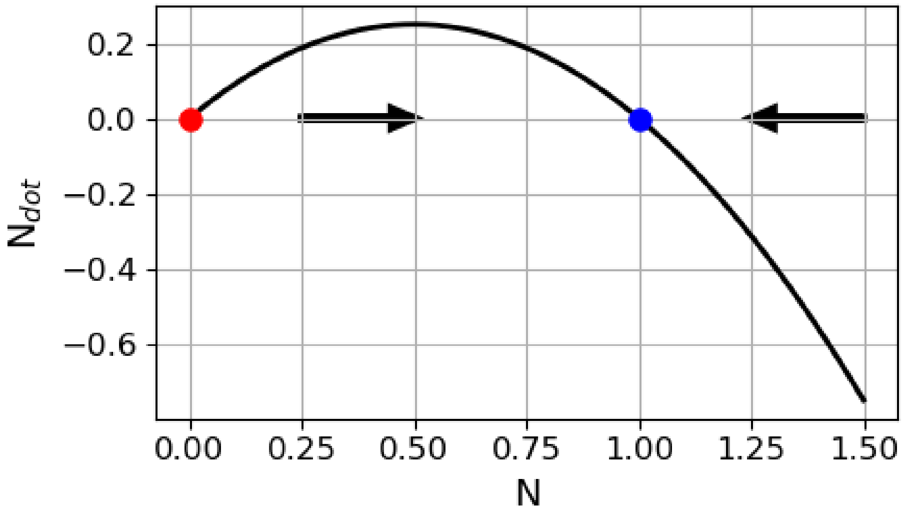

Fig. 2.1. Logical equation for carrying capacity $K = 1$. Fixed points: red dot (unstable) and blue dot (stable). The flow of the population is aways away from an unstable fixed point and towards a stable fixed point.

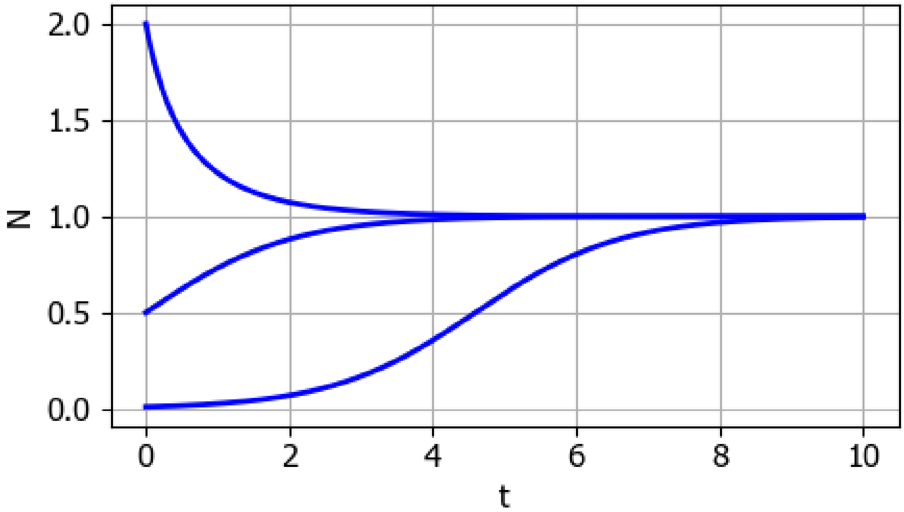

Fig. 2.2. Time evolution of the population for three initial conditions. The three populations converge to the population capacity, *K*.

The logistic equation has been tested in laboratory experiments in which colonies of bacteria, yeast, or other simple organisms were grown in conditions of constant climate, food supply, and absence of predators. These experiments often yielded growth curves with an impressive match to the logistic predictions. However, for organisms with more complicated lifecycles, the agreement between experiment and predictions was not so good.

Another way to explore [1D] dynamical systems is to plot the **slope field** for the system in the (t, x) plane (figure 2.3). The equation $\dot{x} = x(1 - x)$ with $r = 1$ and $K = 1$, can be interpreted in a new way: for each point (t, x), the equation gives the slope dx / dt of the solution passing through that point. The slope field can be show using a quiver plot (figure 2.3) or a streamplot (figure 2.4). Then, finding a solution now becomes a problem of drawing a curve that is always tangent to the local slope.

Fixed points dominate the dynamics of first-order systems. In all our examples, all trajectories either approached a fixed point, or diverged to $\pm\infty$. These are the only things that can happen for a vector field on the real line [1D]. The reason is that trajectories are forced to increase or decrease monotonically, or remain constant. To put it more geometrically, the phase point never reverses. Hence the **impossibility of oscillations.**

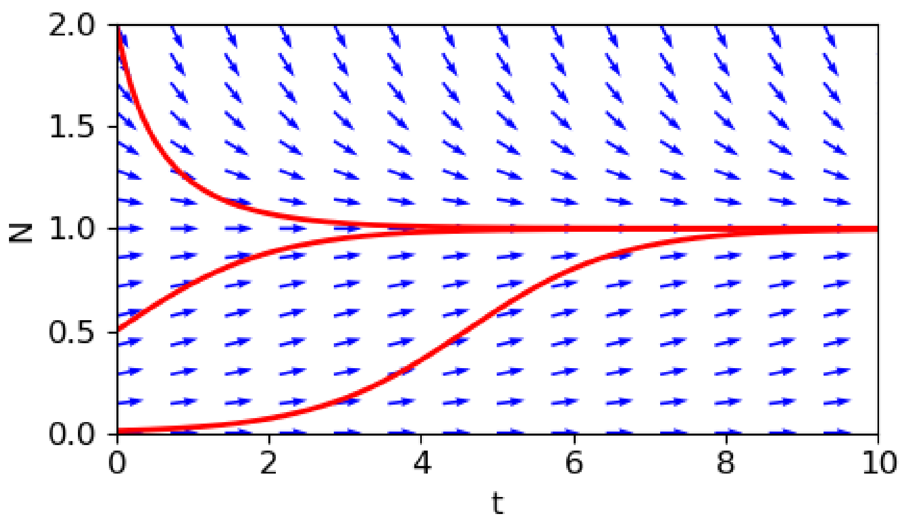

Fig. 2.3. Slope field quiver plot (normalized so that all arrows have unit length). All trajectories converge to the carrying capacity $K = 1$.

The slope is given by dY/dX for dY is the function for $\dot{x}$ and dX is set to 1.

```
N = 15; t = linspace(0,10,N); x = linspace(0,2,N)
f = x*(1-x)
T,X = np.meshgrid(t,x)
dX = np.ones([N,N])
F = X*(1-X)
dY = F/(np.sqrt(dX**2 + F**2))
dX = dX/(np.sqrt(dX**2 + F**2))
```

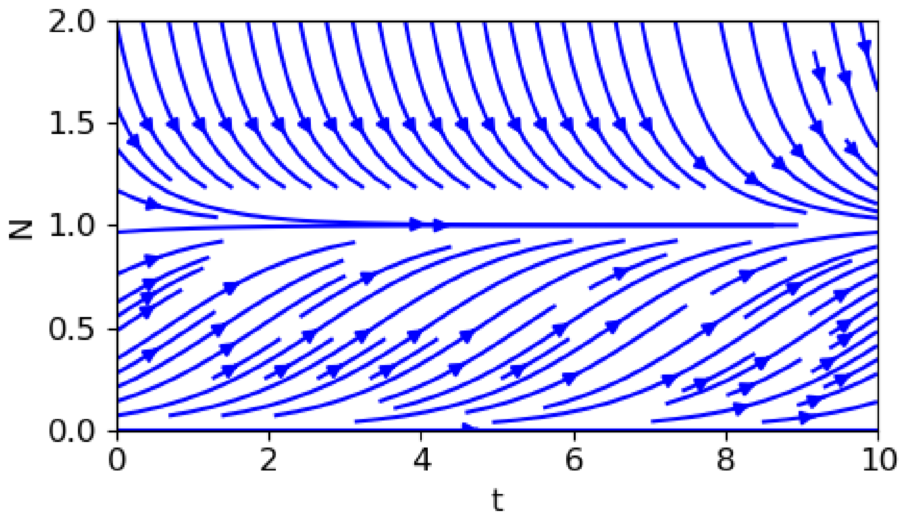

Fig. 2.4. Slope field streamplot. All trajectories converge to the carrying capacity $K = 1$.

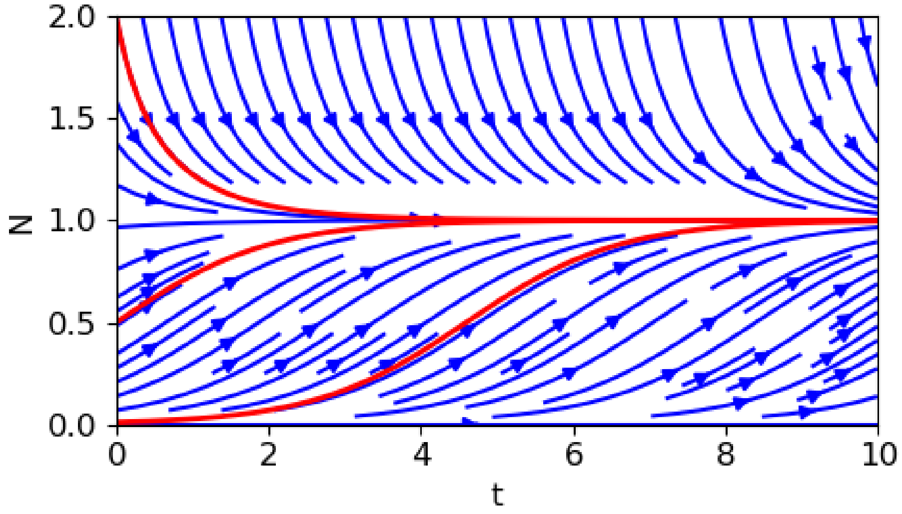

Fig. 2.5. Slope field and three trajectories with different initial conditions.