IB 120/201 - Lab 3

Analytical Solutions to ODEs and Systems of Equations

Due Date: February 12, 2021 by 12:59pm PST

University of California, Berkeley *GSI: Ksenia Arzumanova*

The previous lab introduced you to a numerical approach to solving the logistic growth model. Now we'll cover systems of equations and analytical solutions to ODEs.

Background

Systems of Equations

Performing operations on large and multidimensional datasets involves solving a system of equations. Instead of a simple algebra problem where one must solve for *one* unknown (i.e. $2x + 1 = 7$), we must solve for multiple unknown variables, with more than one equation to guide us. Consider the following example:

$$5x_1 + 3.5x_2 = 4.38 \tag{1}$$

$$x_1 + x_2 = 1 \tag{2}$$

In order to solve this by hand, one can solve for x_1 in terms of x_2 and plug it into equation 1. This is feasible in the simple case of 2 equations and 2 unknowns, but what if we have more than 10 unknowns? Solving by hand all of a sudden seems like a daunting task. It is therefore necessary to employ linear algebra methods and represent the coefficients in matrix format. Now one can solve for $x = \begin{bmatrix} x_1 x_2 \end{bmatrix}$ with the following setup:

$$Ax = b$$

$$A = \begin{bmatrix} 5 & 3.5 \\ 1 & 1 \end{bmatrix}, b = \begin{bmatrix} 4.38 \\ 1 \end{bmatrix}$$

To solve this by hand requires some practice of methods from linear algebra. However, these methods get more complicated the more unknowns you have. This class will not cover manually solving systems of equations in this way, but rather how to implement Python solvers. If you are interested in the theory and magic that happens behind the Python functions, I highly recommend taking a class in Linear Algebra or doing some research.

Differential Equations

Differential equations relate a function to its derivative. For instance, consider the following equation:

$$\frac{dx}{dt} = \alpha x \qquad x(0) = 1$$

This is a way of representing exponential growth - the slope of the function at any given time t is proportional to the value x takes. In order to analytically solve this, we need to separate the variables and integrate:

$$\int \frac{dx}{x} = \int \alpha dt$$

Taking the integral yields the following solution:

$$\ln x + C = \alpha t$$

In order to resolve C, we must employ the initial condition x(0) = 1. Solving for C by plugging in 0 for t and 1 for x, we get C = 0, so the final solution is:

$$e^{\ln x} = e^{\alpha t} \longrightarrow x = e^{\alpha t}$$

Something to remember is that not all differential equations have an analytical solution (cannot be expressed through elementary functions, such as polynomials, rational functions, trigonometric functions, etc.). For this reason, most often you'll need to resort to numerical approximations of solutions.

Equilibrium Points

Equilibrium points of a model are where the model stays doesn't change/stays fixed. Conceptually, this is the same as saying the slope is equal to 0. To find equilibrium points we can set the derivative function of the model and set it to 0:

$$\frac{dx}{dt} = f(x) = 0$$

You can test this by plugging in the equilibrium points into the solution of the differential equation as the initial condition. To tie this back to the quadratic equation problem in the last assignment, the harvest differential equation $\frac{dN}{dt} = rN(t)(1 - \frac{N(t)}{K}) - h$, when expanded, yielded the quadratic equation $rN(t) - \frac{rN(t)^2}{K} - h$. Setting that to 0, in order to find the equilibrium points, you were asked to find those points using the quadratic formula.

Assignment

1. Come up with a system of equations with 3 unknowns and 3 equations (in contrast to the example with 2 unknowns and 2 equations) and solve it using one of the functions covered in lab.

2. Solve the following differential equation using *dsolve*():

$$\frac{dx}{dt} = \alpha x^2 \qquad x(0) = 1$$

3. Write a function that models the predator-prey relationship (Lotka-Volterra model),

$$\frac{dN}{dt} = aN - bNP$$

$$\frac{dP}{dt} = cNP - dP$$

4. Solve it with the following parameters and plot the results: $a = 1.8$; $b = 0.4$; $c = 0.48$; $d = 1$. Set the initial condition to $N_0 = 10, P_0 = 10$.

5. Find the equilibrium points of the model using *nonlinsolve*() (Why *nonlinsolve*() and not *dsolve*()? *nonlinsolve*() solves the nonlinear system of equations, rather than solving the differential equation - we're interested in the points that make the differentials equal to 0.)

6. Verify that these points are the equilibrium points by plugging them into the initial condition.

7. Test the stability of each set of points by adding 0.5 to the equilibrium points in the initial condition statement.