Tutorial 6
Optimization

Learning Objectives

In this tutorial you will explore a model with a system of two nonlinear differential equations as well as a modification on that model.

These problems relate to the following course learning objectives:

- *Model with a system of ODEs.*
- *Analyze the stability of equilibrium points.*
- *Linearize a nonlinear problem around a point of interest.*

Problems

1. Consider the Lotka-Volterra Predator-Prey model

$$\frac{dx}{dt} = ax - pxy \qquad \text{(prey equation)}$$
$$\frac{dy}{dt} = -by + qxy \qquad \text{(predator equation)}$$

with the parameters $a, b, p, q > 0$.

(a) Describe how the system of differential equations models a predator-prey interaction.

(b) Find the equilibrium solutions. How would you label them for a lay person?

(c) Calculate the Jacobian and decide on the stability of each equilibrium solution.

(d) One of the equilibrium solutions is stable, but not asymptotically stable / attracting (solutions starting nearby, don't converge towards the equilibrium, but also don't diverge away).

Let us show this fact:

i. To show this, use the system of ODEs above to find an expression for $\dfrac{dy}{dx}$.

ii. Solve the differential equation you obtained for $y(x)$.

iii. Your solution has the form $E(x, y) = C$ for an arbitrary constant C.

iv. Conclude that the solution of the system of ODEs, "live" on the level sets of a function, which are closed curves. So the system has periodic orbits.

v. Use `python` to visualize the level sets of E.

vi. Simulate a solution using Euler's method and Runge-Kutta's method. What do you observe?

(e) The Lotka-Volterra model has provided insight into real-world ecological phenomenon. Consider the example:

> *A beach-side community became panicked after a string of shark attacks one summer. They decided to aggressively hunt the sharks, which dramatically reduced their numbers near the beach. However, the following year there was a considerable increase in the number of shark sightings.*

Recalling that the sharks are predators of fish, explain the situation.

Use a sketch to describe what probably happened in your level set graph.

2. [1] Now let us consider a modified Lotka-Volterra model to account for logistic growth of the prey population:

$$\frac{dx}{dt} = ax\left(1 - \frac{x}{K}\right) - pxy \qquad \text{(logistic prey equation)}$$
$$\frac{dy}{dt} = -by + qxy \qquad \text{(predator equation)}$$

with $a = b = p = q = 1$ and $K > 0$.

(a) What value of K recovers the original Lotka-Volterra predator-prey model?

(b) Find the equilibrium solutions and the values of K for which they are all valid. How would you label them for a lay person?

(c) Classify the stability of each equilibrium solution.

(d) Sketch a phase portrait and describe the effects of including a logistic prey growth in the Lotka-Volterra model.

[1] Based on the 2024 test 1.

1. (a) Each equation describes how the corresponding populations change.

   For the prey equation:

   - ax means that without the other term (predation), the prey population will grow proportionally to its current number – exponential population growth (Malthusian growth)
   - xy is the number of total possible encounters between 1 prey and 1 predator and
   - p is the probability of a possible encounter actually happening (some prey live too far from some predators) times the probability that when there is an encounter, the predator actually catches the prey.[2]

   For the predator equation:

   - $-by$ means that without the other term (predation), the predator population will start dying (probably of hunger!), and the death rate is proportional to its current number
   - xy has the same meaning as for the prey equation
   - $q = p \cdot r$,m where p is the same as in the prey equation and r is a way to quantify how much a successful hunt (or food) will contribute to breeding more baby predators.

   (b) We nee to solve $\frac{dx}{dt} = \frac{dy}{dt} = 0$. We get two solutions:

   - $x = y = 0$: Extinction
   - $x = \frac{b}{q}, y = \frac{a}{p}$: Co-existence

   (c) The Jacobian is:

$$J = \begin{bmatrix} a - py & -px \\ qy & -b + qx \end{bmatrix}$$

   - $x = y = 0$: we get the eigenvalues $a > 0$ and $-b < 0$, so the equilibrium is repelling and unstable (saddle point)
   - $x = \frac{b}{q}, y = \frac{a}{p}$: we get the eigenvalues $\pm i\sqrt{ab}$, which have real part 0, so we can't conclude the stability from eigenvalue analysis.

   (d) This question focuses on the equilibrium $x = \frac{b}{q}, y = \frac{a}{p}$.

   i. We have

$$\frac{dy}{dx} = \frac{\frac{dy}{dt}}{\frac{dx}{dt}} = \frac{(-b + qx)y}{(a - py)x}$$

   ii. To solve it, we write it as a separable ODE

$$\frac{a - py}{y}\frac{dy}{dx} = \frac{-b + qx}{x}$$

   which we can now solve to get

$$a\log(y) - py = -b\log(x) + qx + C$$

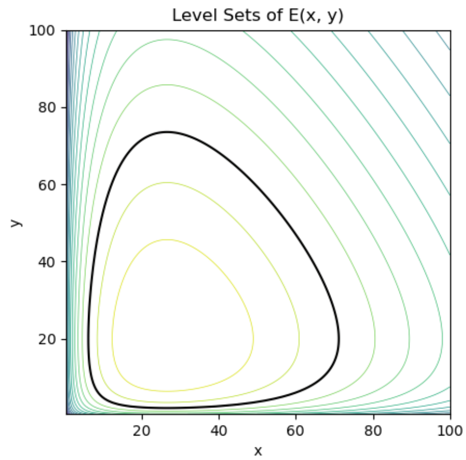
   iii. We can write the solution as

$$E(x, y) = a\log(y) - py + b\log(x) - qx = C$$

   iv. The equation above means that each solution traces the C-level set of $E(x, y)$
   Since E is continuous for $x, y > 0$, we can conclude that the preimages $E^{-1}(\{C\})$ are compact sets.

[2] Success hunting rates on predators are pretty low - see `https://en.wikipedia.org/wiki/Hunting_success`

v. The python code for this exercise and the next one is here: `tutorial6.ipynb`



vi. When plotting these with the same initial conditions, you should be clear that the Euler simulation doesn't give periodic solutions, as the errors make the simulation spiral out. Runge-Kutta however, gives periodic orbits.

2. (a) We can see that to recover the original Lotka-volterra model, we need to "remove" the extra term, which will happen in the limit as $K \to +\infty$.

   (b) The new system has three equilibrium solutions:

   - $x = y = 0$: Extinction
   - $x = K, y = 0$: Predator extinction with ideal prey population
   - $x = 1, y = \frac{K-1}{K}$, for $K \geq 1$: Co-existence

   (c) The Jacobian is

$$J = \begin{bmatrix} -py + a\left(1 - \frac{2x}{K}\right) & -px \\ qy & qx - b \end{bmatrix} = \begin{bmatrix} -y + \left(1 - \frac{2x}{K}\right) & -x \\ y & x - 1 \end{bmatrix}$$

   So for each equilibrium we have:

   - $x = y = 0$: Eigenvalues ± 1, so it is repelling unstable (saddle point)
   - $x = K, y = 0$: Eigenvalues $-1 < 0$ and $-(K-1) < 0$, so it is attracting stable
   - $x = 1, y = \frac{K-1}{K}$:
     $1 < K \leq \frac{1}{2} + \frac{1}{\sqrt{2}}$: Negative eigenvalues, so it is attracting and stable
     $K > \frac{1}{2} + \frac{1}{\sqrt{2}}$: Real part of eigenvalues < 0, so it is attracting stable (spiral sink)

   (d)

Learning Objectives

Students need to be able to...

- *Model with a system of ODEs.*
- *Analyze the stability of equilibrium points.*
- *Linearize a nonlinear problem around a point of interest.*

Context

In lecture we studied some examples of linearizing and studying the stability of equilibrium points. We do so again in this tutorial.

We go in a bit more detail to prove some properties of the Lotka-Volterra model, such as showing that the solutions form periodic orbits.

Before Tutorial

Send an announcement to students letting them know that they will need to bring a laptop and will be using Jupyter Notebooks `https://utoronto.syzygy.ca`.

What to Do

Introduce the learning objectives for the day's tutorial.

Problem 1 is quite long, so there might not be any time for problem 2. That is ok. If that happens, tell the students that they should work on problem 2 on their own.

1.(a) Students should analyze and explain the meaning of each term on the RHS

1.(b) Students will probably be a little stumped here. Ask them, if they had to write a tweet (xeet?) describing each equilibrium solution, what would they write?

1.(c) When we do the Jacobian, we are linearizing the ODEs, so somethings to keep in mind:

- All eigenvalues have real parts < 0 : Attracting / Asymptotically stable equilibrium
- One eigenvalue has real part > 0 : Repelling / Asymptotically unstable equilibrium
- All eigenvalues have real parts $= 0$: Can't tell! (like when the second derivative is 0, we can't tell if it's a max/min/none!)

1.(d) Insist that students do the `python` parts. Tell them to use previous `python` codes (from homework/lecture/...) - for plotting and for numerical methods.

The Euler and Runge-Kutta methods are described in the lecture slides.