MATH 367 – CALCULUS 3 – ASSIGNMENT

Instructions

Complete all the exercises. Write up your solutions neatly and clearly. Include computer generated plots, and relevant code, and a brief explanation of how they were generated. You may work in teams of up to three people, submitting jointly. The assignment is due in class in class on Monday, April 3.

Predator-prey systems

On an isolated island live a population of rabbits and a population of wolves. Let $x(t)$ and $y(t)$ be the numbers of rabbits and wolves on the island at time t, respectively. Our goal is to model $x(t)$ and $y(t)$. As is typical in modelling problems, we model quantities by describing their evolution over time, i.e., their rates of change. Assume that this evolution is governed by the following rules:

(1) The rabbits have an infinite food supply. The rabbit population grows at a rate proportional to its size.

(2) The rabbits are eaten by the wolves at a rate proportional to the sizes of both rabbit and wolf populations. Getting eaten by a wolf is the only way a rabbit dies.

(3) The population of rabbits is the only food source for the wolves. The wolf population grows at a rate proportional to the sizes of both rabbit and wolf populations.

(4) The wolves are mortal; their population declines at a rate proportional to its size.

Exercise 1. Provide justifications for the four assumptions above. Offer a few critiques, too.

By assumptions (1) and (2), there are positive constants a and b such that

$$x'(t) = \begin{pmatrix}\text{rate of growth of}\\ \text{rabbit population}\end{pmatrix} - \begin{pmatrix}\text{rate of decline of}\\ \text{rabbit population}\end{pmatrix} = ax(t) - bx(t)y(t).$$

By assumptions (3) and (4), there are positive constants c and d such that

$$y'(t) = \begin{pmatrix}\text{rate of growth of}\\ \text{wolf population}\end{pmatrix} - \begin{pmatrix}\text{rate of decline of}\\ \text{wolf population}\end{pmatrix} = cx(t)y(t) - dy(t).$$

Thus, our populations are modelled by the following system of first-order differential equations:

$$(*) \qquad \begin{aligned} x'(t) &= ax(t) - bx(t)y(t) \\ y'(t) &= cx(t)y(t) - dy(t). \end{aligned}$$

Exercise 2. Suppose $x(t) = 0$ for all t. What is $y(t)$? Suppose $y(t) = 0$ for all t. What is $x(t)$? Are these answers reasonable?

Exercise 3.

- If $x(t) \geq 0$ and $x'(t) = 0$, then $y(t) =$ ____________. If $y(t) \geq 0$ and $y'(t) = 0$, then $x(t) =$ ____________. Your answers should be lines in the xy-plane. Plot them.
- On the same plot, indicate the following regions:

$$\begin{aligned}
R_{++} &= \{(x(t), y(t)) \in [0, \infty) \times [0, \infty) : x'(t) > 0 \text{ and } y'(t) > 0\}, \\
R_{+-} &= \{(x(t), y(t)) \in [0, \infty) \times [0, \infty) : x'(t) > 0 \text{ and } y'(t) < 0\}, \\
R_{-+} &= \{(x(t), y(t)) \in [0, \infty) \times [0, \infty) : x'(t) < 0 \text{ and } y'(t) > 0\}, \\
R_{--} &= \{(x(t), y(t)) \in [0, \infty) \times [0, \infty) : x'(t) < 0 \text{ and } y'(t) < 0\}.
\end{aligned}$$

- On the same plot, roughly sketch some parametric "population trajectory" curves $(x(t), y(t))$ satisfying $(*)$. Indicate direction along the curves with arrowheads.

Although relatively simple, the system $(*)$ does not have a simple, closed form solution $(x(t), y(t))$, as far as I know. Let's try to solve it numerically. Suppose we are given initial conditions:

$$(x(t_0), y(t_0)) = (x_0, y_0).$$

Let Δt be a small time increment and set $t_1 = t_0 + \Delta t$. By first order approximation,

$$x(t_1) = x(t_0 + \Delta t) \approx x(t_0) + x'(t_0)\Delta t = x_0 + x'(t_0)\Delta t.$$

Assuming that $(x(t), y(t))$ satisfies $(*)$, we have

$$x'(t_0) = ax(t_0) - bx(t_0)y(t_0) = ax_0 - bx_0y_0.$$

Therefore,

$$x(t_0 + \Delta t) \approx x_0 + (ax_0 - bx_0y_0)\Delta t.$$

Exercise 4. Approximate $y(t_1)$ in terms of x_0 and y_0.

Thus, given $(x(t_0), y(t_0))$, we can compute an approximation to $(x(t_1), y(t_1))$. Repeating this process, we can approximate $(x(t_2), y(t_2))$ in using $(x(t_1), y(t_1))$, where

$$t_2 = t_1 + \Delta t = t_0 + 2\Delta t.$$

Continuing in this manner, we get a sequence

$$\big\{(x(t_k), y(t_k)) : k = 0, \dots, n\big\}, \quad \text{where} \quad t_k = t_0 + k\Delta t.$$

Joining consecutive points with line segments gives a piecewise-linear approximation to a parametric population trajectory curve, $(x(t), y(t))$.

Exercise 5. Write a computer program in the language of your choice or make an excel spreadsheet to solve $(*)$ numerically. Present results, including plots, for values of a, b, c, d, x_0, and y_0 illustrating the various possibilities for the long-term evolution of our populations. You may take $t_0 = 0$, for convenience.

Suppose you're a biologist and that you've collected population data for rabbits and wolves populations, i.e., a list of triples (t_k, x_k, y_k), where x_k and y_k are the measured sizes of the rabbit and wolf populations, respectively, at time t_k. Suppose your data is modeled by a parametric curve $(x(t), y(k))$. Then we should have

$$x'(t_k) \approx \frac{x_k - x_{k-1}}{t_k - t_{k-1}} \quad \text{and} \quad y'(t_k) \approx \frac{y_k - y_{k-1}}{t_k - t_{k-1}}.$$

If our model accurately reflects the evolution of the populations under consideration, we should also have

$$x'(t_k) = ax(t_k) - bx(t_k)y(t_k) \approx ax_k - bx_ky_k$$

and

$$y'(t_k) = cx(t_k)y(t_k) - dy(t_k) \approx cx_ky_k - dy_k.$$

We get the approximate identities

$$\frac{x_k - x_{k-1}}{t_k - t_{k-1}} \approx ax_k - bx_ky_k, \qquad k = 1, \dots, n,$$

and

$$\frac{y_k - y_{k-1}}{t_k - t_{k-1}} \approx cx_ky_k - dy_k, \qquad k = 1, \dots, n.$$

We want to find the parameters a, b, c, and d for which the cumulative error in the above approximations is as small as possible. Introduce the following (mean-square) error functions:

$$E_x(a,b) = \sum_{k=1}^{n} \left(\frac{x_k - x_{k-1}}{t_k - t_{k-1}} - (ax_k - bx_ky_k) \right)^2$$

$$E_y(c,d) = \sum_{k=1}^{n} \left(\frac{y_k - y_{k-1}}{t_k - t_{k-1}} - (cx_ky_k - dy_k) \right)^2$$

Exercise 6. Using (a subset of) the data you computed in Exercise 5, try to recover a and b by minimizing $E_x(a, b)$ and c and d by minimizing $E_y(a, b)$. You might use gradient-descent here, although you're free to use another methods as long as justify its applicability.