ESMI Applied Math - Summer 2019 18 Jun 2019

The Lotka-Volterra Model

Expected Skills.
Students can...

- *encode ODE models in MATLAB.*
- *generate trajectories and phase plots of ODE models in MATLAB.*
- *identify and classify invariant features of ODE systems.*
- *perform parameter studies of ODE models in MATLAB.*
- *contextualize mathematical statements to conclusions about the model.*

Guiding Questions

(a) What is the general workflow of implementing an ODE system in MATLAB?

(b) What are some useful built-in functions of MATLAB to generate time simulations?

(c) What are equilibria and how do you go about classifying them?

Encoding Lotka-Volterra Model in MATLAB

The Lotka-Volterra Model is a system of ODEs that is often used to study predator-prey interactions:

$$\dot{x} = \alpha x - \beta xy,$$
$$\dot{y} = \delta xy - \gamma y.$$

- Within the context of predators and prey, what do the variables x and y represent? the parameters $\alpha, \beta, \delta, \gamma$? the equations themselves?
- Encode the ODE: `du = lotkaVolterra(t,u,p)`. What does this function do?
- Encode the Euler method integrator: `[t,u] = forwardEuler(odefun,tspan,u0)`. Explain in your own words what this function does. Recall that a single forward Euler step is determined by the formula:

$$u_{k+1} = u_k + \Delta t \cdot f(t_k, u_k).$$

Analyzing The Lotka-Volterra Model

(a) Set $(\alpha, \beta, \delta, \gamma) = (1, 2, 1, 1)$. Using `forwardEuler` with a step size of $\Delta t = 0.001$, compute an ODE solution with initial condition $(x_0, y_0) = (1, 1)$. Plot the solution as a function of time. What do you notice about the solution and what does this mean about predator and prey populations?

(b) Compute two more ODE solutions with initial conditions $(x_0, y_0) = (0, 1)$ and $(x_0, y_0) = (1, 0)$. What do these solutions mean about predator and prey populations?

(c) Plot the vector field generated by `lotkaVolterra` by using the built-in `quiver` function in MATLAB.

(d) Overlay a phase plot on the vector field by plotting several ODE solutions as trajectories in phase space. What do you notice about the trajectories and the vector field?

(e) There are two special trajectories that are just single points. They are known as *equilibrium points*, defined as states where the model prescribes no change. Can you find them analytically?

(f) [**Parameter Investigation**] How do different values of α change the trajectories and equilibrium points? Can you mathematically determine/describe these changes? What does this represent in your model? Repeat this exercise for β, δ, and γ.

Aside on ODE Integrators

The Lotka-Volterra model has a very interesting property in that it has a *conserved quantity*. To obtain this quantity, we can eliminate time in the ODE and arrive at the single separable equation:

$$\frac{dx}{dy} = -\frac{y(\delta x - \gamma)}{x(\beta y - \alpha)}.$$

(a) Solve this separable equation to obtain the conserved quantity, V.

(b) We can use this conserved quantity to evaluate the accuracy our ODE integrators. With an initial condition of $(x_0, y_0) = (1, 1)$, make plots that illustrates how the time step used in `odeEuler` and how the conserved quantity changes with time.

(c) The Euler method is known as a first-order explicit method since error at a given time is proportional to the step size and uses the state of the system at the current time to calculate the state at a later time. An alternative to this is the first-order implicit method, known as the *backward Euler method*. For this method, we require to solve the implicit equation for u_{k+1}:

$$u_{k+1} = u_k + hF(t_{k+1}, u_{k+1}).$$

Encode `[t,u] = backwardEuler(odefun,tspan,u0)`.

(d) Compute the ODE solution with initial condition of $(x_0, y_0) = (1, 1)$.

(e) In your own words, explain the difference between the two Euler methods and compare their accuracy.

(f) [**Project Idea**] Look through MATLAB's library of ODE integrators like `ode45`. Investigate their derivations, accuracy, and numerical stability.