Dynamical Systems Theory – Getting Started

Here is a brief overview of some important steps to follow when starting with any new system of equations. Our computers can do these steps easily, and you now have code to do this, but it's good to practice these by hand for simple systems (one- and two-dimensions) to help your understanding of these models.

I have underlined important terms throughout this document, and made a list below so that you can review them and begin making your own definition table. This is by no means an exhaustive list of important terms for dynamical systems theory, but is important for these steps described below. We will add to this as we go on.

---

1. Start with the equation(s) describing a system – these describe the rate of change over time, *t*, in density of our state variable(s), or population(s). Let's say we have one resource population; we can call this equation f_R.

2. Set f_R to 0 and solve for R. This tells us when the rate of change is 0, and therefore the population is at an equilibrium. Often we refer to these solutions as R*.

   Note that with multiple state variables (e.g., a C-R model), our solutions will lead to functions (e.g., lines, curves) that depend on the other state variable(s) rather than just fixed points. These are what we call zero-growth isoclines, also referred to as nullclines. Our equilibria (C*, R*) are where these clines intersect, such that the rate of change each state variable is 0.

   - At this point it can be useful to plot these equilibrium solutions in state space. In a one-dimensional case, we can visualize our solutions on a line (one dimension), and in multiple dimensions we plot our clines as functions of our state variables in this space (e.g., solutions of C as a function of R for a two-dimensional C-R model).

   - One we have our solutions plotted, we can immediately see how each of the parameters in our model affect the equilibria and begin to visualize where we may see bifurcations.

   - It is also useful to plot trajectories or vector fields on each of these clines and around our equilibria. Keep in mind that trajectories will only point in one dimension directly on the clines, so note that the geometry of the isoclines importantly affects the dynamics. Visually piecing together the entire phase space immediately tells us a lot about stability before we even move on to the next steps!

3. Next we want to linearize the system by taking the partial derivatives of our equations with respect to each of the state variables. Note that in a one-dimensional system (e.g., f_R), we just need to take the derivative (i.e., df_R/dR). We put this in matrix form such that for every dimension, *n*, of our system we get an *n* x *n* matrix, which we call the Jacobian matrix. For a two-dimensional system (e.g., f_R and f_C), this will be in the form:

$$\begin{pmatrix} {}^{\partial f_R}/_{\partial R} & {}^{\partial f_R}/_{\partial C} \\ {}^{\partial f_C}/_{\partial R} & {}^{\partial f_C}/_{\partial C} \end{pmatrix}$$

Once we have solved for our derivatives, we can substitute in our equilibrium solutions. We can do this for each of our equilibria. (Remember we always have trivial equilibria that still matter!) This results in a linearization of our model around the equilibria.

4. Now we can calculate our eigenvalues. In our one-dimensional case, the work is done for us. Our eigenvalue is simply equal to our derivative (technically this is a 1 x 1 matrix). For multiple dimensions we need to do a bit of matrix algebra to diagonalize our matrix, which effectively transforms our matrix to a form that separates the dimensions of our eigenvalues. This transformation is as follows...

   Every matrix, **A**, has eigenvalues, λ, and eigenvectors, **v**, such that

   $\mathbf{Av} = \lambda\mathbf{v}$

   This can be re-written as follows:

   $\mathbf{Av} - \lambda\mathbf{v} = 0$
   $\mathbf{Av} - \lambda\mathbf{Iv} = 0$
   $(\mathbf{A} - \lambda\mathbf{I})\mathbf{v} = 0$

   Note here that **I** is an *n* x *n* identity matrix, so λ**I** is a *n* x *n* diagonal matrix with $\lambda_1 \ldots \lambda_n$ on the diagonals. (**A** – λ**I**) is our diagonalized matrix.

   Finally, as long as **v** is non-zero, then this will only have a solution if:

   $|\mathbf{A} - \lambda\mathbf{I}| = 0$

   This equation is referred to as the characteristic equation of **A** and is an n^{th} order polynomial in λ with *n* distinct (but not necessarily unique) roots.

Using the characteristic equation of our Jacobian matrix, we can now solve for our eigenvalues associated with each equilibrium. To do this, we take the determinant of our (**A**- λ**I**) matrix. Note that this diagonalization has simply transformed this to be our Jacobian matrix with λ subtracted in the diagonals, but it is good to know where this transformation comes from. Finally we can now use simple algebra to solve the resulting polynomial for λ.

HINT:

For a 2 x 2 matrix, **A**, of the form

$$\mathbf{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

The determinant, det(**A**) or |**A**|, is equal to (ad) - (bc). In this case, the determinant gives us a quadratic equation, which we can solve using the quadratic formula:

If a quadratic equation has the form: $x\,\lambda^2 + y\,\lambda + z = 0$

Then $\lambda = \frac{-y \pm \sqrt{(y^2-4xz)}}{2x}$

** Note that for more information, any linear algebra textbook should explain eigenvalues and eigenvectors (and how to solve for them, including matrix transformations and diagonalization) in more detail.

Definitions

State variable

Parameter

Equilibrium

Trivial equilibrium

Isocline

Nullcline

State space

Phase space

Trajectory

Vector field

Bifurcation

Jacobian matrix

Linearization

Eigenvalue

Eigenvector

Diagonalization

Characteristic equation

Polynomial

Determinant

Quadratic equation

Quadratic formula