Experimental Soft Matter Physics

Module 3: The Logistic Map

Learning Objectives

1. To confirm, using scientific programming techniques, that a simple, *deterministic* equation can generate what appear to be unpredictable, non-repeating sequences of number.
2. To explore structured cascades of period-doubling bifurcations in simple nonlinear systems near a transition to chaos.
3. To appreciate the broader significance of the logistic map in the history of nonlinear science.

Introduction

In your recent exploration of the Lorenz equations, you encountered two different phenomena. First, you found a *bifurcation* in which the steady state behavior of the system suddenly changes: after the control parameter r is increased beyond a critical value, the origin is no longer stable and the system is instead drawn toward a new steady state with nonzero values of X, Y, and Z. This transition echoes the sudden onset of convection in the original fluid dynamical model, from which the Lorenz equations were derived. Second, for much larger values of r, you found a completely different type of behavior: the steady states vanish altogether and the system becomes *chaotic*.

This recognition that a simple deterministic system can exhibit seemingly unpredictable behavior was only the beginning, however. Examining oscillations in the variable $Z(t)$, for example, you found a sequence of peak values, Z_1, Z_2, Z_3 and so on, that follow no obvious pattern. And yet, there is a pattern: given the k-th peak value, you can predict the next peak value using a remarkably simple formula,

$$Z_{k+1} = f(Z_k). \tag{1}$$

Lorenz's original plot of the curve $f(Z)$ is shown at the top of the next page and, in the lab you just completed, you rediscovered this curve for yourselves. Imagine now that you had never seen the original data. What if I simply showed you $f(Z)$, explaining that the entire sequence of Z_k values was governed by a simple formula? Would you expect a sequence generated in this way to be chaotic? In this lab, you will consider these questions using a simplified alternative to $f(Z)$, replacing Lorenz's sharply-pointed empirical curve with a perfect parabola, i.e.,

$$x_{k+1} = a x_k (1 - x_k), \tag{2}$$

where a is a constant. This system, which is known as the *logistic map*, has a long history in disciplines far removed from fluid dynamics, especially population biology. While it seems totally harmless at a casual glance, the logistic map exhibits a striking cascade of bifurcations near a transition to chaos. These features highlight the complexity lurking in simple nonlinear systems, one that provides a useful complement to your exploration of the Lorenz system.

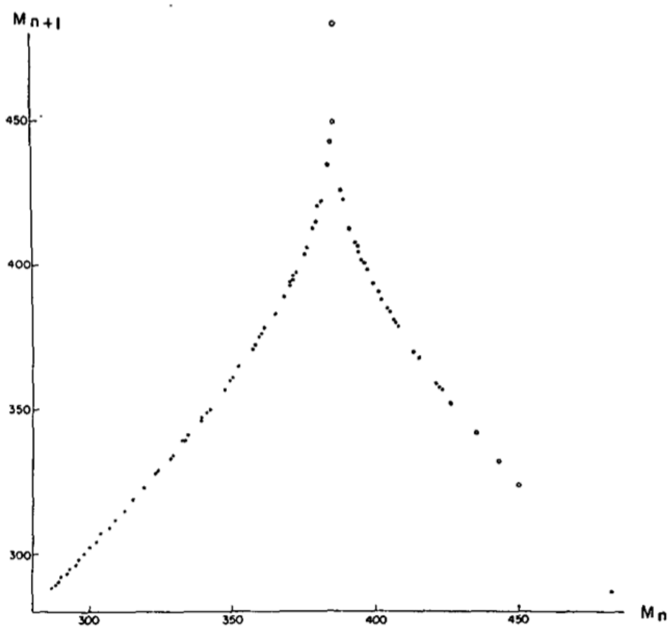

FIG. 4. Corresponding values of relative maximum of Z (abscissa) and subsequent relative maximum of Z (ordinate) occurring during the first 6000 iterations.

Iterating the Logisitic Map

Function of a single variable maps one number to another. If one feeds the output value back into any such function, one is led to another number. Repeating this process n times generates a sequence of n numbers. Mathematicians generally use the term *map* when referring to systems whose dynamics are iteratively defined in this way. Note that these assignments are deterministic in the sense that identical inputs produce identical sequences of numbers.

Investigating maps numerically is, in many ways, much easier than solving a differential equation. Models involving differential equations follow a system's evolution continuously in time and solving such equations on a computer requires carefully controlled approximation techniques. Maps, on the other hand, uses discrete jumps to model the flow of time and require no approximations. All one needs is code that translates a map's iterative algorithm into appropriate commands that the computer can understand. Using what you have learned this semester, write a Matlab function that iterates the logistic map n times for a chosen value of a. Your code with probably look something like the following:

```
function x = logisticmap(a,n)

   % Initialize an array in which to store results
   x = zeros(n,1);

   % Choose first entry randomly
   x(1) = rand;
```

```
    % Loop over entries, using the logistic map to assign subsequent entries
    for k = 1:n-1
       x(k+1) = ...;
    end

end
```

Note how looping over k controls the iteration of the map. (You'll need to complete the line that defines how the $(k+1)$-th value of x is generated from the k-th value.)

Once your code is working, spend some time exploring the sequences it produces. Note that $f(x) = ax(1-x)$ describes an upside-down parabola, a much simpler curve than the one Lorenz discovered empirically in his model. Since the only parameter in the model is a, you only have one knob to turn (though you do also need to provide a finite value for n for the code). Start with a very small but nonzero value of a and see what happens. Does the sequence settle down to a steady state or does it oscillate? If it oscillates between different values, how many of these values are there? Now repeat for a slightly larger value of a, slowly working your way up from nearly zero until you reach 4 (but do not go past 4). Notice anything interesting? Have fun with this! Only after you've already discovered some of the system's surprises on your own should you dive into the reading and see what others have to say about this remarkable system.

The Period-Doubling Cascade and Chaos

After you've gotten a good sense for what the logistic map is capable of, both from your reading and from running your own code, your challenge will be reproducing and exploring the famous *period-doubling* bifurcation diagram associated with this system. This diagram is emphasized very strongly in both Robert May's famous paper and in James Gleick's book, so you may already know what to expect. You also already have some hands-on experience with bifurcation diagrams, having produced one for the Lorenz system centered around $r = 1$, so you should be able to code this for yourselves (though the plotting will not be simple, since you can't just use the final point in each sequence. . . you'll want to plot more than one x_k for each value of a). See if you can't reproduce one or more of the diagrams found in your readings!

If you're interested in digging a little deeper, here are some fun possibilities for further exploration (by no means required):

1. When you look at the bifurcation diagram more closely, you find that there are windows of order in the chaotic regime: there are fairly large values of a at which chaos is replaced with simpler behaviors. Investigate these windows and see how many of them you can find.

2. As Gleick points out, these windows themselves contain replicas of the entire bifurcation diagram, with each branch of the stable oscillation initiating its own miniature periodic-doubling cascade. Find one of these miniature cascades and zoom in on it, mapping it out in enough detail to convince yourself that this really happens.

3. The Lorenz system contains period-doubling bifurcations as well. If you're interested, I'll give you some hints about where to look for them. Alternatively, some famous experiments exploring period-doubling bifurcations in thermal convection were performed in the early 1980's by a number of French scientists. Similar bifurcations have been found in dripping faucets, driven pendulums, and other relatively simple experimental systems. If you dig around online, you'll find some discussion of these systems.