

THE LOGISTIC MAP AND CHAOS

Reminder: We're considering the logistic (or quadratic) map with parameter a :

$$y = f(x) = ax(1 - x), \quad \text{with } 0 < a < 4. \quad (1)$$

Recall that the fixed points for this map are at $x = 0$ and $x_0 = 1 - 1/a$. The derivative of $f(x)$ is $f'(x) = a - 2ax$, and the equilibrium point is stable if $|f'(x_0)| < 1$. Since $|f'(1 - 1/a)| = |2 - a|$, this point is stable if $1 < a < 3$, and unstable otherwise. On the other hand, when $0 < a < 1$, the other equilibrium point $x = 0$ is stable since $|f'(0)| = |a|$.

As you might imagine, $a = 3$ is another bifurcation point. What happens here? In Fig. 1, the first 100 points of the orbit starting at $x_1 = .35$ are plotted. In the first panel, $a = 2.8$, while in the second, $a = 3.1$. What you see is that after an initial *transient* of 20-30 points,

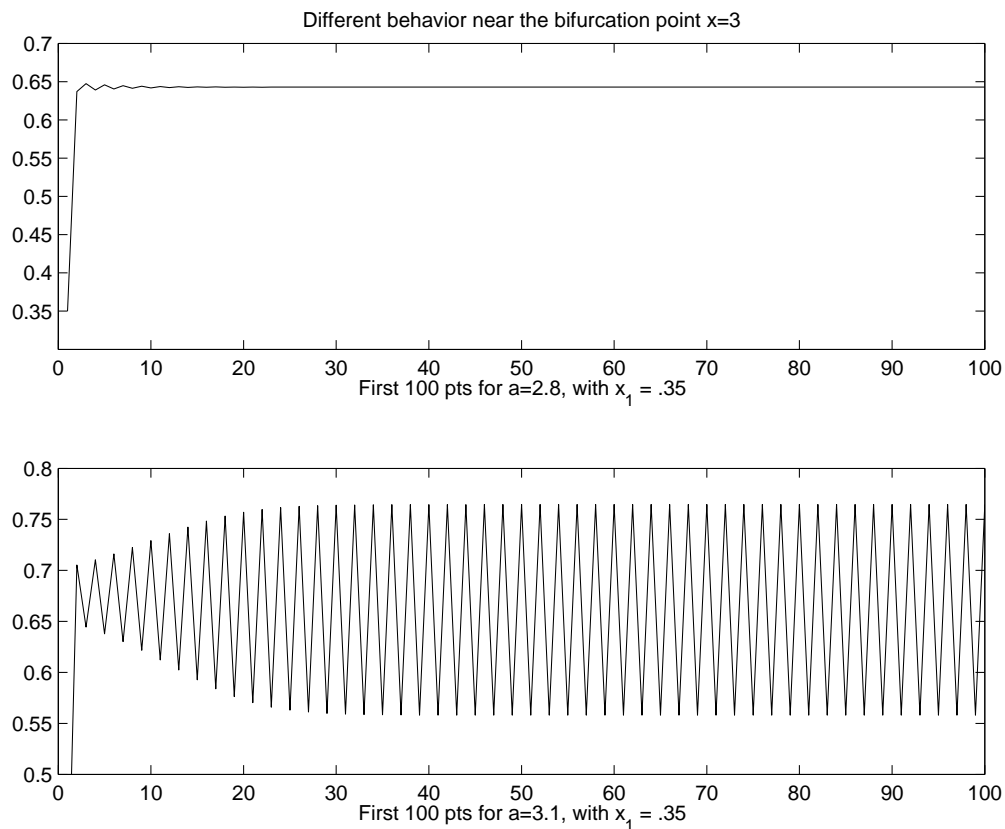


Figure 1: Appearance of an orbit of period 2

both orbits “settle down” into a steady-state. In the first case ($a = 2.8$) the orbit settles onto the stable equilibrium point. You should notice that there’s considerable “roundoff” error here due to the computer. If the arithmetic were exact, the orbit would approach the fixed point arbitrarily closely but never get there in finite time. Also, the display has a finite pixel size, so once the orbit is sufficiently close to the equilibrium point, it’s visually

indistinguishable from it. Also, the duration of the “transient” (as opposed to steady-state) part of the orbit depends on the initial condition.

In the second plot, after the transient, we see that the orbit settles into an orbit of *period 2*, oscillating between (approximately) .55 and .75. This period-2 orbit is *stable* or *attracting* in the following sense: Independent of the initial condition in $(0, 1)$ *every* orbit approaches the period 2 one after the transient, initial behavior.

Summing up, we have, proceeding from left to right on the a axis:

- For $a \in (0, 1)$, the point $x = 0$ is a stable equilibrium.
- There is a bifurcation at $x = 1$; that is, there is a noticeable change in the behavior of the map (1). The point $x = 0$, which is still an equilibrium point, becomes unstable, while the previously unstable equilibrium point at $x_0 = 1 - 1/a$ has now become stable.
- For $a \in (1, 3)$, the point $x_0 = 1 - 1/a$ is stable.
- Another bifurcation point occurs at $a = 3$. The second fixed point of f becomes unstable, but a new stable structure appears, this time an orbit of period 2.

The goal of this lab is to examine what happens as a increases from 3 to 4. Sounds simple enough. Just for your information: at this point, nothing chaotic has occurred. In fact, the behavior of the logistic map for the values of a considered thus far is completely deterministic and non-chaotic in the following sense: orbits which start with neighboring initial conditions actually converge to one or the other of the stable orbits we’ve seen as $n \rightarrow \infty$.

♣ **Exercise(s)**: If there’s an orbit of period 2, then there exist points x_1, x_2 , such that $f(x_1) = x_2$, and $f(x_2) = x_1$. Thus x_1 and x_2 are both fixed points of the map $f^2(x) = f(f(x))$. Of course, any fixed point of f is also a fixed point of f^2 , and any fixed point of f gives an orbit of period 2 as well ($x_1 = x_2$). But usually we mean by an orbit of period 2 a fixed point of f^2 which is not also a fixed point of f .

In the case of the logistic map,

1. What is the map f^2 ?
2. What is the equation satisfied by a fixed point of f^2 ?
3. Since fixed points of f are also fixed points of f^2 , you already know two solutions. That should allow you to factor the equation obtained in (2) and find the other two roots.
4. Of the 4 fixed points of f^2 , which are stable in $a > 3$ and until what value of a do they remain so?

5. What do you think happens at the next bifurcation point?

GRAPHICAL ANALYSIS OF THE ORBITS

A convenient way to visualize what's going on is to plot, instead of the orbit $\{x_1, x_2, x_3, \dots\}$, the sequence of points given by $\{(0, x_1), (x_1, x_1), (x_1, x_2), (x_2, x_2), (x_2, x_3), (x_3, x_3), \dots\}$: start on the x -axis, and then go up to the main diagonal (the line $y = x$); then draw a line connecting this point to the point $(x_1, f(x_1)) = (x_1, x_2)$; then go to the corresponding diagonal point (x_2, x_2) , and so on.

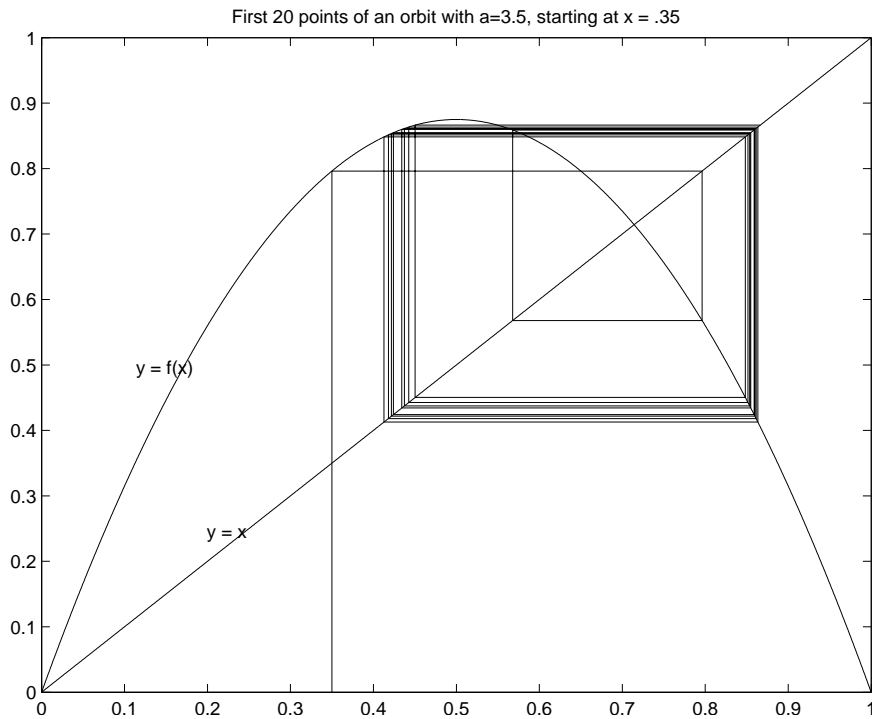


Figure 2: The beginning of an orbit with $a = 3.5$

In Fig. 2, you can see how this goes for the first 20 points of the orbit when $a = 3.5$, and the initial point is $x_1 = .35$. It's not clear what's actually going on here in terms of stability, since we're still in the transient region for this orbit.

The figure was produced using the MatLab function "iterate", the full script for which appears at the end of these notes. The usage of this function is

```
>> [u,v,z] = iterate(a, x1, n);
```

where a is the value of the parameter, x_1 is the initial point on the orbit, and n is the number of iterations to compute. The output of the function is three vectors: z consists

of the successive points on the computed orbit (for example, Fig 1 was made using the command `>> plot(z)`. The other two vectors u, v are the sets of ordered pairs that go with the graphical analysis mentioned above. Fig. 2 was made using the following commands:

```
>> [u,v,z] = iterate(3.5,.35,20);
>> x = 0:.01:1; (makes 100 equally spaced points in [0,1])
>> y = logistic(3.5, x); (computes  $y = f(x)$  vectorially, for the logistic map, with  $a =$ 
>> plot(u,v);
>> hold (keeps the current plots on the figure)
>> plot(x,y); (draws the graph of  $f$ )
>> plot(x,x); the diagonal.
>> hold (releases the plot, so the figure can be used for something else,
the next time you plot)
```

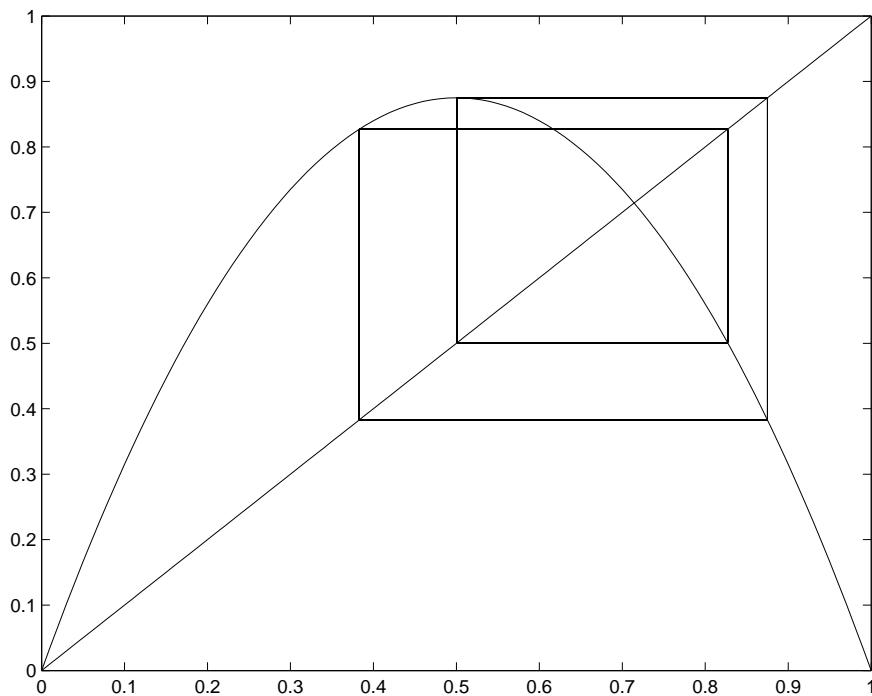


Figure 3: The stable part of the previous orbit

In Fig 2, it's not clear how long the transient solution lasts. The easiest thing to do to find the steady-state part is to compute lots of points on the orbit and throw away most of them. Fig. 3 was produced using the following

```
>> [u,v,z] = iterate(3.5,.35,1000);
>> whos
Name          Size          Bytes  Class  Attributes
```

```

u          1999x1          15992  double
v          1999x1          15992  double
z          1001x1           8008  double
>> plot(u(1950:1999), v(1950:1999))

```

From this, it's clear graphically that when $a = 3.5$, there's a stable orbit of period 4. The points on the orbit can be extracted from the variable z :

```
>> z(900:910)
```

```
ans =
```

```

0.8750
0.3828
0.8269
0.5009
0.8750
0.3828
0.8269
0.5009
0.8750
0.3828
0.8269

```

Evidently, the orbit consists of the 4 points $\{x_1 = 0.8750, x_2 = 0.3828, x_3 = 0.8269, x_4 = 0.5009\}$, with $f(x_1) = x_2, \dots, f(x_4) = x_1$.

You might wonder if we've missed a stable orbit of period 3 somewhere between $a = 3.0$ and $a = 3.5$. The answer is "no".

♣ **Exercise(s)**: Show that, for $a = 3.3$ there is no orbit of prime period 3. Hint: you can demonstrate this graphically by checking out the fixed points of $f^3(x)$. There are just two fixed points of $f^3(x)$, and both correspond to the unstable fixed points of f itself, which of course give degenerate (and therefore not prime) period 3 orbits. Compare this result with the results you get from plotting f^4 for the same value of a .

As you may have surmised by this point, as a increases, what occurs is often called the *period-doubling cascade*. That is, as a increases, the period 4 orbit will become unstable, and a stable orbit of period 8 will appear; this will be followed by one of period 16, and so on. This all occurs in a rather small region of the parameter a , and there are other things that happen as well by the time $a = 4$ is encountered.

CHAOTIC BEHAVIOR IN THE LOGISTIC MAP

Suppose we consider another initial condition, say $x_1 = 0.7$ for the case $a = 3.5$. In Fig 4, we plot the first 50 points for the ICs $x_1 = 0.35$, and $x_1 = 0.7$. It can be shown that this

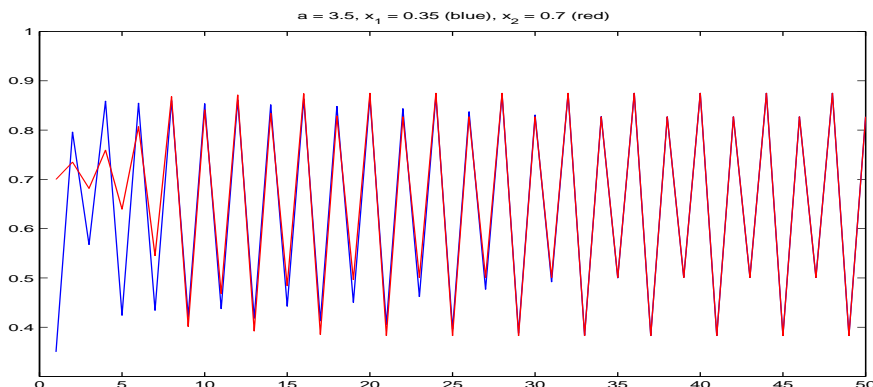


Figure 4: The 2 orbits both converge rapidly to the stable period 4 orbit.

convergence occurs for *any* initial condition in the interval $(0, 1)$. On the other hand, consider two nearly identical ($x_1 = 0.7$, $x_1 = 0.701$) initial conditions for the parameter value $a = 4$: Here is our first example of what's called *chaotic* behavior, which is often thought of as

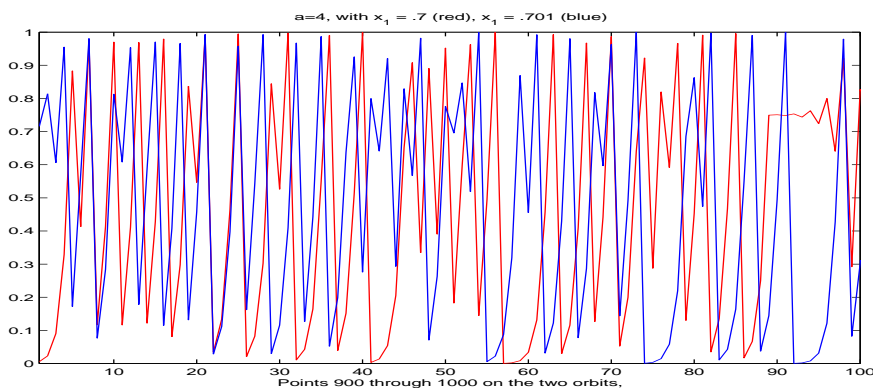


Figure 5: In this case ($a = 4$), the orbits are completely uncorrelated, even after discarding the first 900 points.

“sensitive dependence on initial conditions”. In this case, even though the orbits are nearly identical at the start, after 100 points or so, there's no way to detect, either statistically or by looking at the figure, any such correlation between the two orbits. By contrast, in the previous figure, the two orbits are completely correlated after only 50 iterations.

THE BIFURCATION DIAGRAM FOR THE LOGISTIC MAP

The general behavior of the logistic map depends critically, as we've seen, on the value of the parameter a , and we'd like to produce a graphic that captures the change in behavior as a function of a . So what we do is:

- Pick a value of a in $(0, 4)$.
- Pick an initial condition $x_1 \in (0, 1)$. The results should not depend on x_1 .
- Iterate the logistic map for long enough to get rid of transients, e.g., 1000 points.
- Plot the last 50 points of the orbit against the value of a . So, for instance, if there's a stable period 4 orbit for this value of a , there will be only 4 distinct points in this list of 50. If you plot all 50, you'll only get 4 distinct ones. The following figure shows what happens.

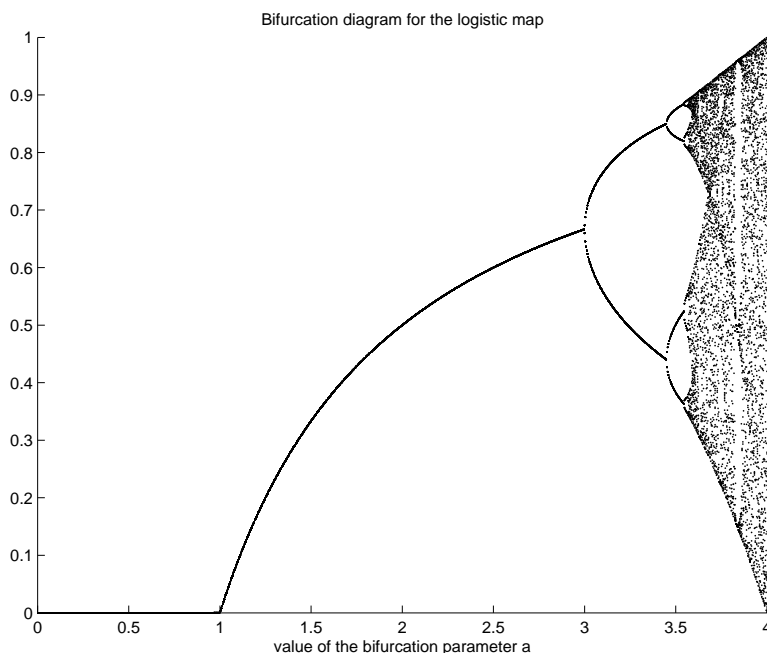


Figure 6: The bifurcation diagram for $y = ax(1 - x)$, $0 \leq a \leq 4$

The MatLab script used to produce this figure (*bifurcate.m*) is listed, along with other scripts you may find useful, in the Appendix. Notice the following things about this (imperfect) figure:

- The figure is just fine for $a \leq 3.5$; it becomes somewhat murky for larger values.
- For each value of a , what's plotted are the points lying on the stable orbit (if any). Thus for $a = 3.2$, there are 2 points, while for $a = 3.5$, there are 4. You can just discern the onset of the period 8 orbit somewhere around $a = 3.6$.

- If there's no stable orbit at the point in question, there will be essentially 50 random points in $(0, 1)$ scattered over that value of a .
- You can improve things by
 - taking a smaller parameter window - e.g., you could modify the script so that you sampled 1000 points equally spaced between 3.5 and 3.75, where it looks as though interesting things are happening.
 - When you do this, you might wish to examine, for instance, the region between 0.75 and 1 on the vertical axis. For this you'll need to display more than 50 points total for each given value of a , or things will be too sparse to see any patterns.
 - If you really magnify some region of the horizontal axis, you may need to take more than the “usual” 1000 points for each value of a .
 - You could further modify the script(s) so that you could specify some interval on the a -axis and some interval on the horizontal axis as part of the input to the procedure. This would involve converting *bifurcate.m* to a MatLab function, rather than a procedure.

♣ Exercise(s):

1. Modify the script *bifurcate.m* to “blow up” diagram, in the smaller window of $3.5 \leq a \leq 3.6$. Describe what you see. Can you locate the approximate onset of any orbits of period higher than 8?
2. You should just be able to see what's called the “period 3 window” in Fig 6. Locate it as well as you can by magnifying the picture as described above. Print a figure similar to Fig 3 displaying an stable period 3 orbit. Say something qualitative about what seems to happen when one moves past this window.
3. Can you locate any smaller rectangles in which the same figure as is seen in Fig 6 reappears?

There is an enormous literature on the subject of chaotic dynamics (chaos is also encountered in the solutions of differential equations, as we'll see later on in the course). For an easy introduction, you could read the non-technical paperback by James Gleick, called “Chaos”.

Sample MatLab scripts: These should be “callable” from your main directory; if for some reason they’re not, just copy them into your home directory. These are “bare bones” scripts, with no error-checking, and just enough comments so that you can use them. They don’t necessarily need to be improved, but they could be. If you do improve them, hand the improved scripts or functions in with the rest of the lab. Note that the percent (%) sign denotes a comment and is not executed.

```
function y=logistic(a,x);
% this takes as input the parameter a and the value of x and
% returns y = ax(1-x). The value of x can be a vector, in which
% case, the corresponding y vector is returned

% Usage: >> x = 0:.01:1; (100 equidistant points in [0, 1])
%           >> y = logistic(3.2, x);
y=a*x.*(1-x);

function [u, v, z] = iterate(a, y0, n)
% Outputs everything needed for graphical analysis of the logistic map; the
% inputs are the parameter a for the logistic map, the starting value y0,
% and the number of iterations for the function to do

% Usage:: to interate 500 times with IC y0 = .32, and parameter .5, we type
% >> [u,v,z] = iterate(0.5, 0.32, 500); to plot the 400th to 450th
% points of the orbit, you’d do
% >> plot(z(400:450)). To do the "web plot", with the last 50 points, type
% >> plot(u(900:1000),v(900:1000))

z = zeros(n+1,1);
z(1) = y0;
for i = 2:n
    z(i) = logistic(a,z(i-1));
end
u = zeros(2*n,1);u(1) = y0;
v = zeros(2*n,1);v(1) = 0;
for i=2:2*n
    u(i) = z(ceil(i/2));
end
v = [0;u(3:2*n)];
u = u(1:2*n-1);
```

```

function [z] = iterate2(a, y0, n)
% this one just computes the orbit, so we don't need u and v of iterate.m
z = zeros(n+1,1);
z(1) = y0;
for i = 2:n
    z(i) = logistic(a,z(i-1));
end

(bifurcate.m)
% this is a procedure and not a function. It is invoked by simply typing
% the word 'bifurcate' at the MatLab prompt. It calls the function
% iterate2, which, in turn, calls the function logistic. The way it's set
% up, it computes the stable orbits (of period <= 50), for 1000 values of
% the parameter a in the interval [0,4]. If you're going to experiment,
% it's easier if you just use 100 values of a to make sure you're doing
% things correctly.
clf
hold
for m=1:1000
    a = 0.004*m;
    z = iterate2(a,0.35,1000);
    plot(a, z(950:1000),'.','MarkerSize',3)
end

```

Some pointers on printing:

>> print -deps Fig4 (this prints the active figure to a file in the current directory named "Fig4.eps", where the ".eps" stands for encapsulated postscript. On the lab machines, you may want to convert this to a pdf file. The syntax (in a terminal window) is \$ ps2pdf Fig4.eps.

>> print -depsc Fig4 prints the active figure (in color) to the file Fig4.eps.

>> hold on (allows you to do multiple plots on the same figure). E.g., `hold on; plot(z(1:100)); plot(z1(1:100),'r')` will plot both, with the second in red.

>>hold off (means the next plot will clear the current figure before plotting)