Problem Set Number 08, (18.353/12.006/2.050)j MIT (Fall 2024)

Rodolfo R. Rosales (MIT, Math. Dept., room 2-337, Cambridge, MA 02139)

December 3, 2024

Due: last day of classes, Fall 2024 (turn it in via the canvas course website).

Contents

1 Liapunov vs Lyapunov vs Ljapunov . . . Which one?

If you search the web, you will find, mostly Lyapunov being used. However, if you look at dynamical system books (e.g.: see the books in the syllabus), what you find is (mostly) Liapunov. For example:
Liapunov used by Strogatz,
Liapunov used by Wiggins,
Liapunov used by Guckenheimer and Holmes,
Liapunov used by Bergé, Pomeau, and Vidal.
Liapunov used by Drazin,
Ljapunov used by Peitgen, Jürgen, and Saupe.
Lyapunov used by Parker and Chua, *Practical Numerical Algorithms for Chaotic Systems* — this one is not in the syllabus.
Why is this? I think it has to do with the ambiguity in translating from the Cyrillic to the Latin alphabet. Technically, none of the above is correct, because the actual name of the person we are referring too (a.k.a. Alexander Liapunov) was written in Cyrillic; something like **Миха́йлович Ляпуно́в**

---

2 Liapunov exponents for 1-D maps #03

Statement: Liapunov exponents for 1-D maps #03

Task#1 of 3. Compute the Liapunov exponent, and make a plot analog to figure 10.5.2 in Strogatz's book (i.e.: example 10.5.3) for the 1-D maps $x_{n+1} = f(x_n,\, r)$ below. In all cases *justify the selected ranges for* x *and* r.
Meaning of "justify". Show that if x_n is in the x-region, then so is x_{n+1}. Thus it makes sense to iterate the map.

1. The **Max-μ map** $f(x) = r\,(1 - |2\,x - 1|^{\mu})$, with $\mu = 1.5$. Range $0 \le r \le 1$ and $0 \le x \le 1$. In addition, make plots for the refined regions:
   (1a) $0.7930 \le r \le 0.8170$ and $0 \le x \le 1$.
   (1b) $0.9236 \le r \le 0.9261$ and $0 \le x \le 1$.
2. **Max-μ map** $f(x) = r\,(1 - |2\,x - 1|^{\mu})$, with $\mu = 2.5$. Range $0 \le r \le 1$ and $0 \le x \le 1$. In addition, make plots for the refined regions:
   (2a) $0.925400 \le r \le 0.928400$ and $0 \le x \le 1$.
   (2b) $0.928078 \le r \le 0.928097$ and $0 \le x \le 1$.
   (2c) $0.928095 \le r \le 0.928099$ and $0 \le x \le 1$.

General remarks and background needed for Task#2.

These maps behave in a fashion similar to the Logistic map $f(x) = 4\,r\,x\,(1-x)$, with a series of bifurcations as r grows, at values: $0 < r_1 < r_2 < \cdots < r_\infty < 1$.

At $r = r_1$ the origin becomes unstable, and a new (stable) critical point $0 < x_* < 1$ is born past it.

At $r = r_2$ a (stable) period two solution is born when x_* becomes unstable (flip bifurcation).

At $r = r_3$ a period doubling occurs, followed by further period doublings at each $r = r_n$, $3 < n < \infty$.

At $r = r_\infty$ a transition to chaos occurs.

Beyond $r = r_\infty$ more bifurcations occur, with periodic "windows" (each including within it a period doubling cascade) interspaced with chaotic regions.

In each interval $r_n < r < r_{n+1}$ the (global) attractor is a periodic solution [1] of period 2^{n-1}. At a particular point $r_n < r = s_n < r_{n+1}$ in each interval the attractor is **super-stable** — meaning that perturbations evolve according to $\epsilon_{n+1} = O(\epsilon_n^2)$. Furthermore:

The r_n converge geometrically to r_∞: $R_n = (r_{n-1} - r_{n-2})/(r_n - r_{n-1})$ is approximately constant.

The s_n converge geometrically to r_∞: $S_n = (s_{n-1} - s_{n-2})/(s_n - s_{n-1})$ is approximately constant.

The Feigenbaum number characterizes the convergence rate; i.e.: $r_\infty - r_n$ and $r_\infty - s_n$ behave like $(\delta_{\text{map}})^{-n}$ for $n \gg 1$.

$$\delta_{\text{map}} = \lim_{n\to\infty} R_n = \lim_{n\to\infty} S_n \qquad (2.1)$$

Note that *the r_n are difficult to compute accurately,* because they correspond to neutrally stable solutions. On the other hand, *the s_n are relatively easy to compute accurately,* since they correspond to super-stable solutions. Hence the second equality in (2.1) provides the *better way to compute* δ_{map}.

How does this all relate to the Liapunov exponent $\lambda_e(r)$?

Chaos yields $\lambda_e > 0$.

Period doubling: r_n is a local maximum of λ_e, where $\lambda_e = 0$. That is: $\lambda_e(r_n) = 0$ and $\lambda_e < 0$ on each side of r_n.

Super-stable attractors yield $\lambda_e = -\infty$. However, *in a numerical calculation this will, generally, not be true.* Instead, downward (negative) spikes in the plot of λ_e versus r (centered at $r \approx s_n$) are seen.

Task#2 of 3. In both cases above for the **Max-μ map** ($\mu = 1.5$ and $\mu = 2.5$), calculate a few s_n (say, for $1 \le n \le 4$) and use these values to get an estimate for $\delta_{\text{Max-1.5}}$ and $\delta_{\text{Max-2.5}}$. Note that 3 significant digits for s_n, $2 \le n \le 4$, will allow you to get S_4 with about 2 significant digits.

Task#3 of 3 (optional). The plot for **(2c)** shows the transition from period doubling to the "beyond r_∞" chaos region — as seen from point of view of the Liapunov exponent. **What does the plot for (1b) show?**

Hints, process, and remarks.

h1. The process to compute the Liapunov exponent λ_e is explained in example 10.5.3 of the book by Strogatz. Specifically: for a given value of r, select a random $0 < x_1 < 1$. Then iterate the map, $x_{n+1} = f(x_n,\, r)$, for $1 \le n \le n_b$, where $n_b \gg 1$ (say: a few thousand times) — this so the attractor is reached. Next continue the map iteration, and start the computation of λ_e, as follows: (i) Define $\lambda_1 = 0$. (ii) Let $\lambda_{m+1} = \lambda_m + \frac{1}{n_p}\log(|f'(x_n,\, r)|)$, for $1 \le m \le n_p$, where $n = n_b + m$, $n_p \gg 1$ (a few thousands, at least), and $f' = \mathrm{d}f/\mathrm{d}x$. (iii) Then $\lambda_e \approx \lambda_{n_p+1}$.

This has to be done for each value of r in a grid within the desired range $r_1 \le r \le r_2$. For example, an equi-spaced uniform grid, with separation $\Delta r = (r_2 - r_1)/N$.

h2. If you do this with MatLab, it is important that you "vectorize" the calculation (for speed). Place all the values of r in an array $\vec{r}$, with the corresponding values for the iterates (and Liapunov exponent approximation) in arrays $\vec{x}_n$ and $\vec{\lambda}_m$, and compute everything simultaneously — note that, at any stage in the iterations, you need to keep (at most) values for two n and two m. Finally: use the MatLab command "print -dpng FigureName" to save the figure as a small png file — this is more reliable than trying to use the GUI in the figure window.

h3. Calculation of the s_n. Since the S_n involve differences of s_n, significant digits are lost in the calculation. To get s_n accurate enough, you will need a fine grid in r (small Δr). However, even with a laptop, a "vectorized" MatLab calculation with $N \sim 10^5$ values of r will take, at most, a few minutes. The main limitation in (2.1) is that, in principle, you need "large" values of n to accurately compute δ_{map} — but you are not asked to do this.

[1] For $n = 1$ the attractor is actually critical point — i.e.: period = 1.

Finally, note that you can get the required values of the s_n graphically, by using "zoom" and the "data cursor" in the MatLab figure window with the plot of λ_e versus r.

3 Nonlinear stability of a discrete map, and flip bifurcation

Statement: Nonlinear stability of a discrete map, and flip bifurcation

Consider a 1-D map, $x_{n+1} = f(x_n)$, where f is smooth. Assume a fixed point $x_f = f(x_f)$, where $f'(x_f) = -1$ — hence linearization does not determine the stability of x_f. Without loss of generality, assume $x_* = 0$, and write

$$f(x) = -x + a\,x^2 + b\,x^3 + O(x^4), \tag{3.1}$$

where a and b are constants. These are **your tasks:**

t1. **Find condition on a and b** that determines wether $x = 0$ is a stable or unstable fixed point. *Hint:*

t1.a The condition looks like: stability if $h(a,\, b) > 0$, and instability if $h(a,\, b) < 0$, for some function h.

t1.b Consider what happens upon iterating $g(x) = f(f(x))$, which you can ascertain by expanding g to $O(x^4)$, using (3.1). Then note: if $x_{2\,n+2} = g(x_{2\,n})$ decays/grows, then so does $x_{2\,n+3}$, because f is continuous.

t2. **Answer this question:** *why do you have to expand g up to $O(x^4)$, in item* t1.b*, to determine stability?* Note that here I expect the mathematical/technical reason for this.

t3. Let a and b in (3.1) be such that $x = 0$ is stable, i.e.: $h(a,\, b) > 0$, and take a map F such that

$$F(x) = -(1+\delta)\,x + a\,x^2 + b\,x^3 + O(x^4), \tag{3.2}$$

where $0 < \delta \ll 1$. Then x is a linearly unstable fixed point, and a **period two (stable) solution**

$$x^*_{n+2} = x^*_n, \quad x^*_{n+1} = F(x^*_n), \tag{3.3}$$

appears,‡ where x^*_n has size $O(\sqrt{\delta})$.
This is called a **supercritical (or soft) flip bifurcation.**

‡ Argument: the same we made to explain the scaling behind supercritical pitchfork and Hopf bifurcations. The new solution appears as a balance between the destabilizing linearity, and the stabilizing nonlinearity.

Your task. Pick an example F where this happens, with $a \neq 0 \neq b$, and show a numerically computed picture of cobwebs† converging to the period two stable solution.

† Use two cobwebs (with different colors), one converging from "inside" and the other from "outside".

I suggest that you write a "generic" program for $F(x) = -(1+\delta)\,x + a\,x^2 + b\,x^3$ and initial data x_0, and then play with the parameters till you get a pretty picture. Further: choose your colors well; e.g.: yellow on a white background is a bad idea! Note: something like $1 < a < 2$, $b \sim -2/3$, and $\delta \sim 0.3^2$, worked for me.

THE END.