University of Zurich[UZH]

Course Work

'The Physics of Life'

Spring Semester 2021

Logistic map and Ricker model: implementation in python

Supervisor:

Dr. Jordi Bascompte

Autor:

Varvara Stein

18-743-500

June 14, 2021

Abstract

Almost all living systems are non-linear dynamic systems and as it has been revealed chaos is capable of describing some of them; especially the systems that are highly sensitive to initial conditions. However, in this case, the chaotic systems are deterministic. For instance, logistic map and Ricker model refer to such systems. They are seemingly simple but, at the same time, they produce unpredictable and divergent results. Therefore, it might be interesting to simulate both of these models and examine their behaviour.

Content

Introduction

This course work aims to implement and visualise logistic map, to define how it is influenced by Lyapunov exponent, and to compare the logistic model with Ricker model. According to Mitchell Feigenbaum's discovery, all systems that undergo period-doubling path to chaos obey a mathematical constant the distance between consecutive bifurcations along the horizontal axis shrinks by a factor that asymptotically approaches 4.669 which is called Feigenbaum's constant. Regardless of the system's specific dynamics, the ratio of the bifurcations on its road to chaos always obeys this constant. This phenomenon can be referred to as universality that implies that very different systems can exhibit very similar behaviour regardless of their underlying dynamics [Boeing, 2016]. Therefore, I would expect them to have a very similar bifurcation diagram. This speculation will be empirically tested in the implementation section.

Another question that I am interested in is the dependence on initial condition; therefore, I would like to test to what extent the models under study are influenced by the initial conditions.

And finally, I would like to carry out a small experiment related to the perturbation term.

Logistic map and Ricker model are simple one-dimensional discrete models describing population dynamics [Braverman, 2013]. Despite the fact, that these models can be relatively easily implemented, their behaviour is far from being simple. Thus, quite complex phenomena, such as loss of stability and a period-doubling route to chaos, can be observed.

Implementation

The models are implemented in python with the help of two libraries. They are NumPy to execute calculations and matplotlib to create plots. Besides, I used official documentation of these libraries and StackOverflow to solve the errors that occurred during the coding.

First, I defined the parameters of a system that I planned to simulate. The size of the system is 1000, the growth rate varies from 2 to 4, and the initial condition equals 0.0001. In this snippet, I also initialised a vector needed for the implementation of Lyapunov exponent.

```
t = 1000 #number of values to simulate, same amount for r, n, and Lyapunov exponent
r = np.linspace(2, 4, t) #place r values in the linear space between 2 and 4
n = 0.0001 * np.ones(t) #initial condition
lyapunov_exp = np.zeros(t) #initialise vector for lambda values
```

Growth rate has a crucial role in the way the population changes over time; therefore, it might be interesting to have a closer look at it. To investigate this parameter, I plotted eight different growth rates (from 0.5 to 4). The example below demonstrates the implementation of the growth rate of 0.5.

```
time_steps = 20
x = np.zeros(time_steps +1)
y = np.zeros(time_steps +1)
x[0], y[0] = 0, 0.4
r = 0.5

for i in range(time_steps):
    y[i+1] = r * y[i] * (1 - y[i])
    x[i+1] = x[i] + 1

fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(1, 1, 1)
ax.plot(x, y, alpha=0.5, color='b', label='r=0.5')
ax.set(xlabel='Generation', ylabel='Population')
plt.legend(loc='upper left',  bbox_to_anchor=(1, 0.5), title='Growth rate')
plt.grid(color = 'grey', linestyle = '--', linewidth = 0.5)
plt.show()
```

As the result, I obtained the following plot:

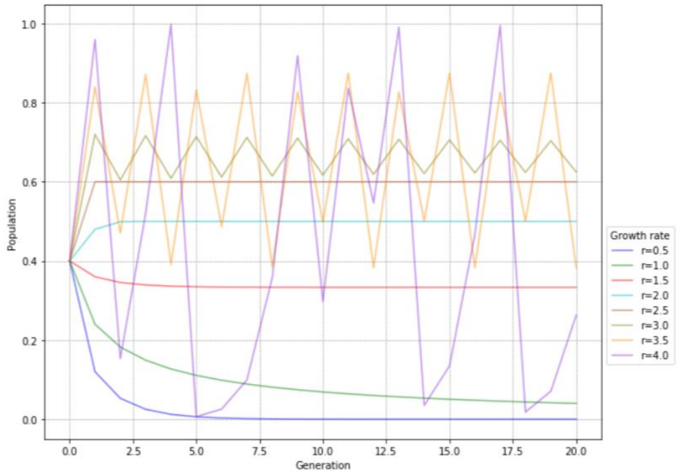


Figure 1. Growth rate

The plot demonstrates that if the growth rate equals 0.5 (or other value below 0), the population dies out; the growth rates 1, 1.5 result in the decrease of population, stable population is reached with the growth rates 2 and 2.5. Growth rates 3 and 3.5 yield a more interesting pattern. As it can be seen in the plot, the population fluctuates that is, it returns to its starting point not every year (after one iteration). In the case of the growth rate of 3, it takes 2 iterations and a rate of 3.5 requires 4 iterations. Nevertheless, such systems still can be called periodical. Unlike them, growth rates greater than 3.5 (in this case, rate=4) exhibit different behaviour and are called aperiodic. In other words, they have a cycle but its period is infinite. Taking into account the peculiarities of the growth rate, it is especially interesting to see a visualisation of the system for the growth rates in the range of 3.5 and 4, as they are expected to be aperiodic.

As a next step, I implemented the formulas of logistic map and Ricker model as two separate functions. The logistic equation is formulated in the following way:

$$x_{t+1} = rx_t(1 - x_t)$$

So, the code in python is quite intuitive; r represents growth rate and n stands for population size at time t.

```
def logistic_map(r, n):
#define the function of logistic map
    return r * n * (1 - n)
```

Ricker model is defined with the following formula:

$$x_{t+1} = \ x_t \, e^{r\left(1-\frac{x_t}{K}\right)}$$

Where x_t represents the population size at time t, r corresponds to the growth rate, and K is the carrying capacity of the environment (equilibrium population density). The formula can be simplified with the substitution $x_t \ = \frac{x_t}{K}$, so it will be defined as:

$$x_{t+1} = \ F(x_t) = x_t \, e^{r(1-x_t)}$$

The implementation in python is done in the following way which also seems to be quite intuitive. As in the previous function r is the growth rate and n is the population size.

```
def ricker(r, n):
#define the function of Ricker model
    return n * np.exp(r*(1 - n))
```

As a next step, I simulated the systems and visualised them in a bifurcation diagram. I ran a simulation for a system across 1000 growth rates between 2 and 4 and iterated over it 1000 times. But for the sake of better visualization, the diagram only shows 100 last iterations. Here I also implement Lyapunov exponent and plot it in the same graph.

```
fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(1, 1, 1)
ax.set_xlim(2, 4)
ax.set_ylim(-2, 2)
ax.axhline(y=0.0, color='k', linestyle='-', lw=.7)
ax.grid(True)
ax.set_title('Lyapunov exponent and logistic map')

for i in range(steps):
    n = logistic_map(r, n) #implement logistic map
    #print(n)
    if i >= (steps - steps_to_show):
        ax.plot(r, n, ',k', alpha=0.25)

    lyapunov_exp += np.log(abs(r - 2 * r * n))#calculate Lyapunov exponent

ax.plot(r, lyapunov_exp/steps, 'g', linewidth = 1)


ax.set_xlabel('growth rate')
plt.tight_layout()

#print(len(lyapunov_exp))
```

The plot illustrates the development of the system in relation to its growth rate. At growth rate 3 the system corresponds to a period-2-cycle (in term of the dynamics of population, possible population splits into two discrete paths) but as the growth rate increases, the system loses its stability and turns in a period-4-cycle (at $r \approx 3.48$), which is then replaced with a stable period-8-cycle (at $r \approx 3.54$) and so forth till the system reaches the critical point of a period of infinity when it becomes aperiodic and chaotic; however, the system only appears to be random. In fact, it follows the same structure and patterns over an infinite period.

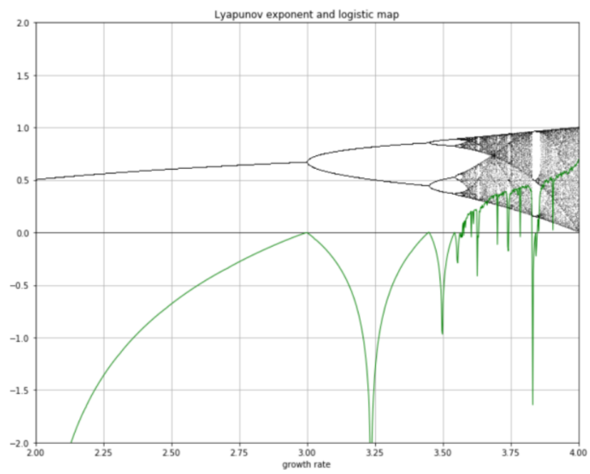


Figure 2. Logistic map and Lyapunov exponent

Apart from bifurcation diagram of logistic map this graph also illustrates Lyapunov exponent marked by a green line.

The term Lyapunov exponent, named after the Russian mathematician Aleksandr Mikhailovich Lyapunov, indicates the speed with which two initially close dynamics diverge - if the Lyapunov exponent is positive - or converge - if the Lyapunov exponent is negative - in phase space [Sayama, 2015]. That is Lyapunov exponent measures how rapid a dynamic system tends to evolve to deterministic chaos.

The plot demonstrates that before the system becomes aperiodic every time Lyapunov exponent approaches zero, the change in cycle occurs. For instance, becomes a period-2-cycle. The diagram shows another important property of Lyapunov exponent, i.e., it is positive when the system is chaotic. Here Lyapunov exponent ascends above zero if the growth rate $r \approx 3.6$. It is also quite interesting that even when the system transforms to chaos, at some points Lyapunov exponent is below zero. These points coincide with periodic windows. So, as the name suggests they can exhibit periodic behaviour. This idea is also supported by the fact that Lyapunov exponent remains below zero.

Then, I compared the bifurcation diagram of the logistic map with the representation of the Ricker model. To plot it, I used the same code with the exception that instead of the function `logistic_map` I applied the function `ricker`.

The output of the script is given below. This diagram proves the speculation that logistic map and Ricker model should have a similar representation. As it can be seen from the graph, as the growth rate increases, the system transforms from period-2-cycle to period-4-cycle, etc. and finally, becomes chaotic or, in other words, period-doubling bifurcations lead to chaos. Similar behaviour was observed in the diagram of the logistic map.

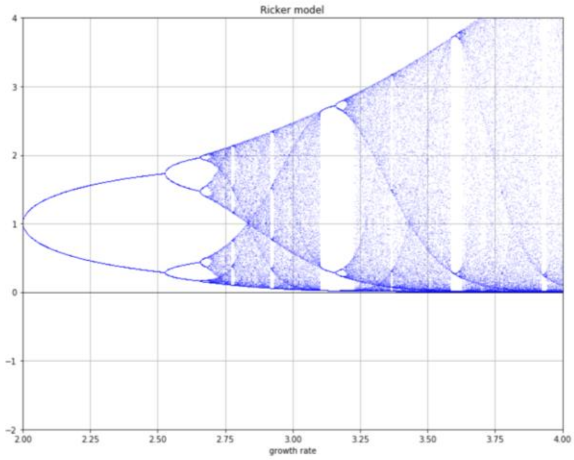
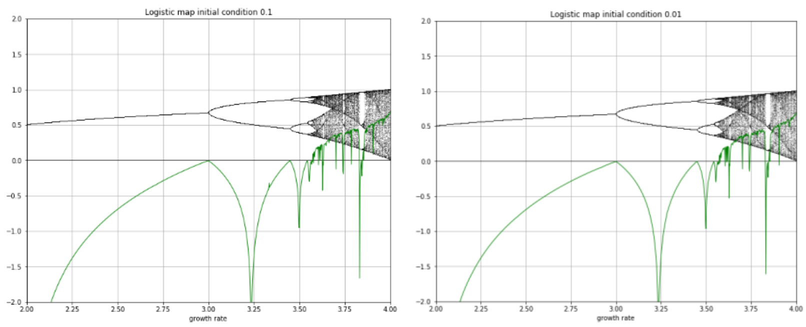


Figure 3. Ricker model

Chaotic systems are usually characterized by the Sensitive Dependence on Initial Conditions (SDIC). Or in other words, they are highly influenced by the initial conditions. For instance, a fine difference in initial conditions may result in drastic divergence in the system. To explore this property, I ran the simulation on logistic map with different initial conditions using the same implementation as before:

$$n_1 = 0.1, n_2 = 0.01, n_3 = 0.001, n_4 = 0.0001$$

As the result, I obtained the following plots:



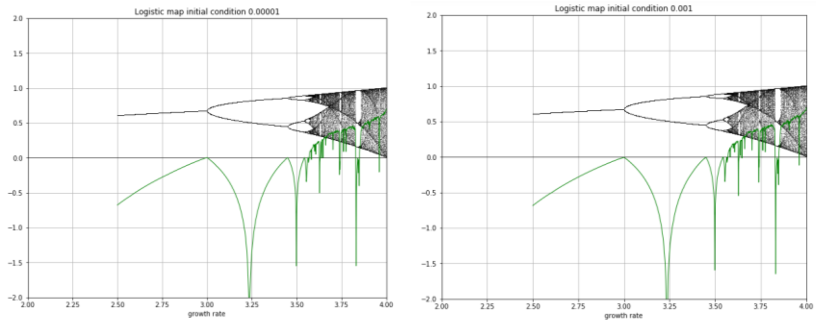


Figure 4. Logistic map with different initial conditions

The diagrams seem to preserve their overall shape regardless the initial condition; however, one can notice that Lyapunov exponent slightly varies which may testify to the fact that models vary as well. Perhaps, it is possible to get a better understanding if regions of chaos are zoomed in.

It is interesting to note that Ricker model can be extended by adding a perturbation term u. This term may be in some cases essential to model the behaviour of many populations; for instance, populations where individuals are either added (positive perturbations) to or harvested from the population (negative perturbations) at each time step [Stone, 1993; Hughes, 2021]. Thus, according to McCallum [McCallum, 1992], if $u > 0$ the system demonstrates a stable two-cycle orbit if values of the growth rate are sufficiently large.

However, if the perturbation term is added to logistic equation, it yields quite a different result that is, remains chaotic.

To obtain proof of this finding, I plotted the modified functions of logistics map and Ricker model. To extend the formulas I used the following implementation (logistic map is modified the same way, i.e., `np.float(u)` is added to the initial formula, where u is the perturbation term). In both cases, I took the value $u = 0.4$.

```
def ricker_added_perturbation_term(r, n, u):
#define the function of Ricker model, perturbation term added
    return n * np.exp(r*(1 - n)) + np.float(u)
```

The plots demonstrate that, indeed, Ricker model converges to a 2-cycle model, whereas the overall shape of bifurcation diagram of logistic map does not significantly change, although it shifts to the left. However, I plotted the functions, adding an arbitrary perturbation term only in range of 0 and 1. There have been studies dedicated to a negative perturbation term in Ricker model, but this case seems to require a different method of implementation.

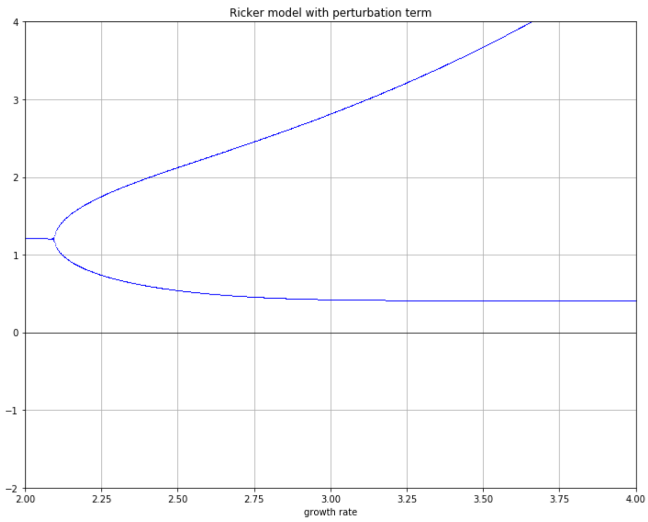


Figure 5. Ricker model and perturbation term 0.4

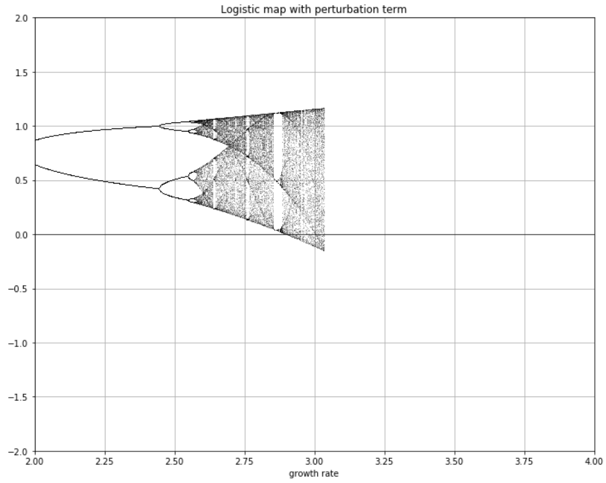


Figure 6. Logistic map and perturbation term 0.4

The initial aim of the work was to examine two dynamical models and to find out if they have similar patterns of behaviour. After the implementation and visualization have been carried out, it is safe to say, that logistic map and Ricker model have quite a number of similarities; however, they are not identical (for example, behave differently if perturbation term I added).

References

Boeing, G. Visual Analysis of Nonlinear Dynamical Systems: Chaos, Fractals, Self-Similarity and the Limits of Prediction. Systems, 4 (4), 37. (2016) https://doi:10.3390/systems4040037

Braverman, E., Rodkina, A. Difference equations of Ricker and logistic types under bounded stochastic perturbations with positive mean. Computers & Mathematics with Applications. Volume 66, Issue 11, 2281-2294. (2013). ISSN 0898-1221. https://doi.org/10.1016/j.camwa.2013.06.014

Hughes, M., Braverman, E. Ricker Discrete Dynamical Model with Randomized Perturbations. (2021)

McCallum, H.I.Effects of immigration on chaotic population dynamics. J. Theoret. Biol. 154 (3) (1992), 277-284

Sayama, H. Introduction to the Modeling and Analysis of Complex Systems. (2015) ISBN 13: 9781942341093

Stone, L. Period-doubling reversals and chaos in simple ecological models. Nature 365, 617–620 (1993). https://doi.org/10.1038/365617a0

Topazh, A. G., Abramova, A.V., Tolstopjatov, S.E. Diskretnye modeli populjacionnoj dinamiki: dostoinstva, problemy i obosnovanie. (2016). Komp'juternye issledovanija i modelirovanie, tom 8, vypusk 2, 267–284 https://doi.org/10.20537/2076-7633-2016-8-2-267-284 (Discrete models of populational dynamics: advantages, disadvantages and justification)