Logistic Map and Chaotic Dynamics

Kexin Liu, Zhenghao Yang, Yunzhao Li, Erdong Zhuang, Lingling Zhu

December 3, 2024

Abstract

The logistic map is a simple mathematical model exhibiting rich dynamical behaviors, including chaos. This report investigates its fixed points, period doubling bifurcations, Feigenbaum constant, and the onset of period-three behavior, with the implications of the Yorke-Li theorem.

Introduction

The logistic map, defined as:

$$f(x) = rx(1-x),\ 0 \le x \le 1,\ 0 < r \le 4$$

is a simple yet profound mathematical model that demonstrates how complex behaviors can emerge from deterministic rules. As the parameter r increases, the logistic map transitions from stability to periodicity and eventually to chaos, making it a foundational tool in chaos theory and nonlinear systems.

The logistic map serves as a gateway to understanding chaotic and nonlinear dynamics, bridging abstract mathematics with real-world applications. It reveals the connection between stability, periodicity, and chaos, demonstrating how deterministic systems can display seemingly random behavior.

1 Fixed Points and Periodic Points in Logistic Mapping

1.1 Introduction to Fixed Points

A fixed point in a dynamical system represents a steady state or equilibrium where the system remains unchanged through mapping. For logistic mapping, which is often adopted to model population growth with resource constraints, a fixed point corresponds to a stable population size that remains unchanged over time. Mathematically, a fixed point x^* satisfies the condition:

$$f(x^*) = x^*,$$

indicating that the mapping of x^* returns the same value.

Substituting the logistic function into this definition, we obtain $rx^*(1-x^*) = x^*$, from which we derive the equation $x^*(rx^* - r + 1) = 0$. By combining the given condition $0 \leq x \leq 1$ with this equation, we conclude that:

- When $0 < r \leq 1$, $x^* = 0$.
- When $1 < r \leq 4, x^* = 0$ and $\frac{r-1}{r}$.

1.2 Introduction to n-Period Points

In contrast to fixed points, n-period points represent states where the system alternates among n distinct values rather than stabilizing at a single point. For instance, in population models, the population size may cycle through n levels over subsequent generations in a predictable pattern. Mathematically, an n-period point x^* satisfies:

$$f^n(x^*) = x^*,$$

where f^n denotes the n-fold composition of f. Fixed points are evidently n-period points with $n = 1$.

1.3 Derivation of 2-Period Points

The analysis of 2-period points in logistic mapping demonstrates varying solutions depending on the value of r. For 2-period points, we solve the equation $f(f(x^*)) = x^*$, which represents the condition where the system returns to its initial value after two-fold composition. Substituting the logistic function $f(x) = rx(1-x)$, we derive:

$$f(f(x^*)) = r\big(rx^*(1-x^*)\big)\big(1 - rx^*(1-x^*)\big) = x^*.$$

The first solution is $x^* = 0$. For $x^* \neq 0$, dividing through by x^* and simplifying yields:

$$r^2\big(rx^{*2} - rx^* + 1\big) = \frac{1}{1-x^*}.$$

Solutions to this equation can be investigated through graphs. As shown in Figure 1,

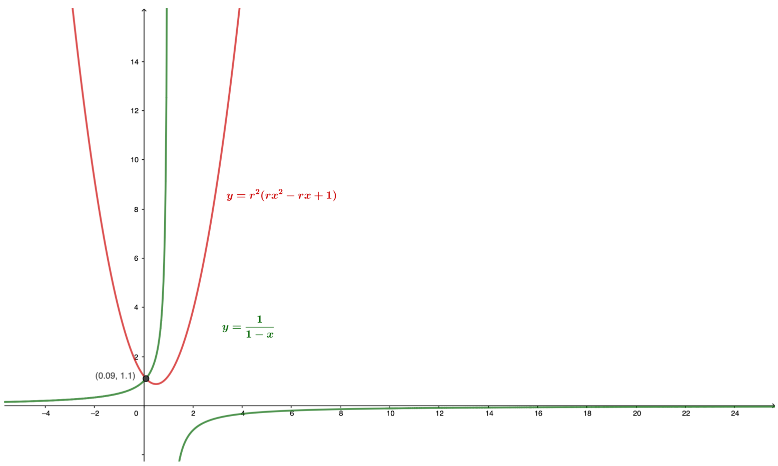


Figure 1: The graph of $y = r^2(rx^2 - rx + 1)$ (red curve) and $y = \frac{1}{1-x}$ (green curve) for $r = 1.1$.

when r is small, the graph indicates that this equation has only one solution. According to **Theorem 9.1**[1], this solution corresponds to the fixed point $x^* = \frac{r-1}{r}$.

The tangency point of these two functions is $x^* = \frac{r-1}{r}$, where their slopes are equal when $r = 3$. At this value, two functions touch but do not cross, resulting in a single intersection. As r exceeds 3, the tangency breaks, leading to the emergence of two additional intersections, which appear on either side of the original tangency point. For

[1]Theorem 9.1: If p is a period-n point, then $f^{kn}(p) = p$ for any positive integer k. Moreover, if j is not a multiple of n, then $f^j(p) \neq p$.

example, Figure 2 illustrates the case where $r = 4$. The graph clearly shows three intersections, indicating three solutions, $x_2^* = \frac{r-1}{r}$, x_3^*, x_4^* to the equation. Indeed, through the help of online calculation programs, one can easily find that $x_3^* = \frac{r-\sqrt{r^2-2r-3}+1}{2r}$ and $x_4^* = \frac{r+\sqrt{r^2-2r-3}+1}{2r}$.

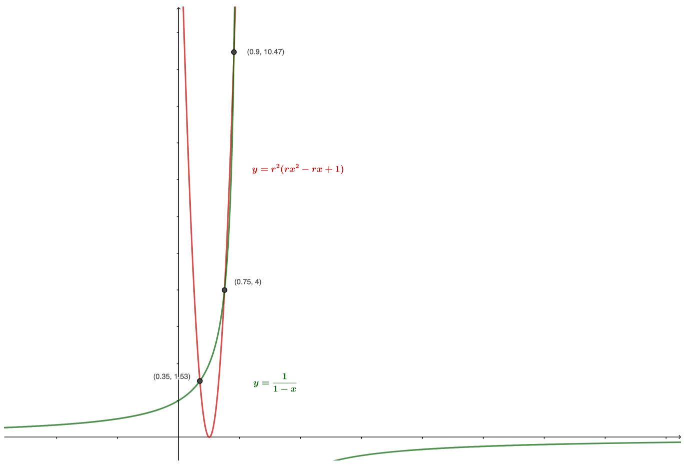


Figure 2: The graph of $y = r^2(rx^2 - rx + 1)$ (red curve) and $y = \frac{1}{1-x}$ (green curve) for $r = 4$.

Therefore, we have reached the conclusion that:

- When $0 < r \leq 1$, the 2-period point is $x_1^* = 0$.
- When $1 < r \leq 3$, the 2-period points are $x_1^* = 0$ and $x_2^* = \frac{r-1}{r}$.
- When $3 < r \leq 4$, the 2-period points are $x_1^* = 0$, $x_2^* = \frac{r-1}{r}$, x_3^* and x_4^*.

1.4 Stability of n-Period Points

The stability of n-period points provides valuable information of the behavior of dynamical systems. An n-period point is said to be **attracting** if the system eventually returns to the n-period point after experiencing a slight disturbance. Formally, this condition is satisfied when the absolute value of the derivative of the n-fold mapping at the n-period point is less than 1:

$$|(f^n(x^*))'| < 1.$$

In contrast, an n-period point is **repelling** if any small disturbance causes the system to diverge from the point, potentially leading to oscillations or chaotic behavior. This occurs when:

$$|(f^n(x^*))'| > 1.$$

When $|(f^n(x^*))'| = 1$, the behavior is more complex, and the point can be attracting, repelling, semistable, or show none of these properties.

1.4.1 Fixed Point Stability

To analyze the stability of fixed points, we consider the derivative of the logistic map $f(x) = rx(1-x)$, which is:

$$f'(x) = r(1-x) - rx = r(1-2x).$$

At $x^* = 0$, the derivative is $f'(0) = r$. Therefore, the stability depends on the value of r as follows:

- When $0 < r < 1$, $|f'(0)| = |r| < 1$, indicating that $x^* = 0$ is **attracting**.
- When $1 < r \leq 4$, $|f'(0)| = |r| > 1$, indicating that $x^* = 0$ is **repelling**.

At $x^* = \frac{r-1}{r}$, the derivative is $f'\left(\frac{r-1}{r}\right) = 2 - r$, and the stability varies as follows:

- When $1 < r < 3$, $|f'(\frac{r-1}{r})| = |2-r| < 1$, indicating that $x^* = \frac{r-1}{r}$ is **attracting**.
- When $3 < r \leq 4$, $|f'(\frac{r-1}{r})| = |2-r| > 1$, indicating that $x^* = \frac{r-1}{r}$ is **repelling**.

1.4.2 2-Period Point Stability

To study the stability of 2-period points, we start with identifying the 2-fold composition for logistic map:

$$f(f(x^*)) = r(rx^*(1-x^*))(1 - rx^*(1-x^*)),$$

and its derivative is computed as:

$$(f(f(x^*)))' = r\left[r(1-2x^*)(1-2rx^*(1-x^*))\right].$$

For $x_1^* = 0$, $(f(f(0)))' = r^2$. Therefore:

- When $0 < r < 1$, $x_1^* = 0$ is **attracting**.

- When $1 < r \leq 4$, $x_1^* = 0$ is **repelling**.

For $x_2^* = \frac{r-1}{r}$, $(f(f(x_2^*)))' = (2-r)^2$. Therefore:

- When $1 < r < 3$, $x^* = \frac{r-1}{r}$ is **attracting**.

- When $3 < r \leq 4$, $x^* = \frac{r-1}{r}$ is **repelling**.

When $r > 3$, two additional 2-period points x_3^* and x_4^* appear. At this stage, both $x_1^* = 0$ and $x_2^* = \frac{r-1}{r}$ become **repelling**. The two new 2-period points are located symmetrically on either side of $x^* = \frac{r-1}{r}$, as shown in the phase portrait.

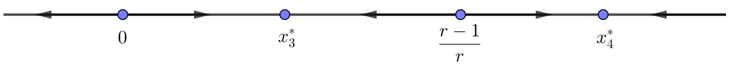


Figure 3: Phase Portrait

For $3 < r < R$ (where $R < 4$), these new 2-period points, labeled x_3^* and x_4^*, are **attracting**.

2 Bifurcation Diagram

To find bifurcations, we first locate the fixed points. From the preceding section, the fixed points are:

$$x^* = 0 \quad \text{or} \quad x^* = 1 - \frac{1}{r} \quad (r > 1).$$

As r increases, the system undergoes a series of bifurcations:

- For $r \in [0, 1]$: $x^* = 0$ is the only stable point, the population becomes extinct.

- For $r \in (1, 3)$: A non-zero fixed point $x^* = 1 - \frac{1}{r}$ emerges and is stable. The number of solutions is 1, which is 2^0.

  Similarly, as previously mentioned, by finding solutions for $f(f(x)) = x$, we obtain two solutions:

$$x = \frac{r + 1 \pm \sqrt{(r-3)(r+1)}}{2r}$$

The system starts its first bifurcation at $r = 3.449$. The number of solutions is 2, which is 2^1.

- For $r \in (3, 3.57)$: The fixed points become unstable, and the system enters period-doubling bifurcations. The system transitions from period-2 (two attractors) to period-4, then period-8, with the period doubling continuously.

  Extending this, by nesting additional f functions around the previous ones, the number of solutions keeps increasing. The number of solutions is given by 2^n, where n is the number of f functions nested around $f(x)$. The equation can also be written as $F(x_n) = x_{n+1} = rx_n\,(1 - x_n)$

- At $r \approx 3.57$: The system enters a chaotic state. Periodic attractors disappear, and the trajectory becomes extremely sensitive to initial conditions.

- For $r \in (3.57, 4]$: The system remains in a chaotic regime, but periodic windows are embedded within chaos. At specific values of r, such as $r \approx 3.83$, stable periodic attractors reappear.

- Bifurcation Diagram shown below. The line $x = 0$ always exists in the diagram. The bifurcation diagram only shows the attracting part, and the repelling part is not shown in the diagram.For example, the line $x = 1 - \frac{1}{r}$ exists in the model but may not be displayed in the diagram.

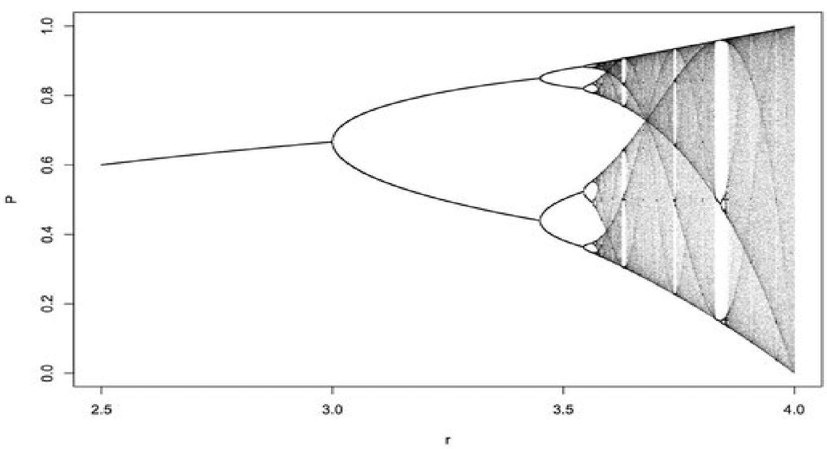


Figure 4: Bifurcation Diagram

3 Period Doubling and Feigenbaum Constant

If we observe the bifurcation diagram of the logistic map, we will find that it shows the phenomenon of period-doubling cascade.

Period-doubling cascade refers to the phenomenon where, in a nonlinear system, the period gradually doubles during the process of the system transitioning from a stable periodic state to a chaotic state. As the parameter r increases slightly beyond $r \approx 3.449$, point 1 and point 2 no longer attract other points, and 4 points with a period of 4 will appear. Other points will be attracted by one of these 4 points and then oscillate between these 4 points. As r continues to increase, there will be 8 points with a period of 8, 16 points with a period of 16, 2^m points with a period of 2^m attract other points..., and this process continues. For any large positive integer m, there will be 2^m points with a period of 2^m. This is the period doubling phenomenon: when the parameters change, the period of the system may show a doubling trend, that is, the originally stable period gradually splits into multiple stable points with smaller periods.

3.1 Example: Period-2 Points

Taking a period-2 point as an example, when the period is two, consider $r = 3.2$. We draw a time series graph when $r = 3.2$. The Logistic Map Time Series Graph is a graph that shows the dynamic behavior of the Logistic Map by numerical iteration, plotted with time step n as the horizontal axis and population density x_n as the vertical axis. It can help observe the behavior of the Logistic Map under a given parameter r. In this case, we find that the graph is jagged, indicating that x_n oscillates back and forth between two values: 0.5 and 0.8 instead of converging to a certain point.

Treat two iterations as one iteration and calculate the relationship between x_{n+2} and x_n.

$$x_{n+2} = rx_{n+1}\left(1 - x_{n+1}\right) \tag{1}$$

$$x_{n+2} = r\left(rx_n(1 - x_n)\right)\left(1 - rx_n(1 - x_n)\right) \tag{2}$$

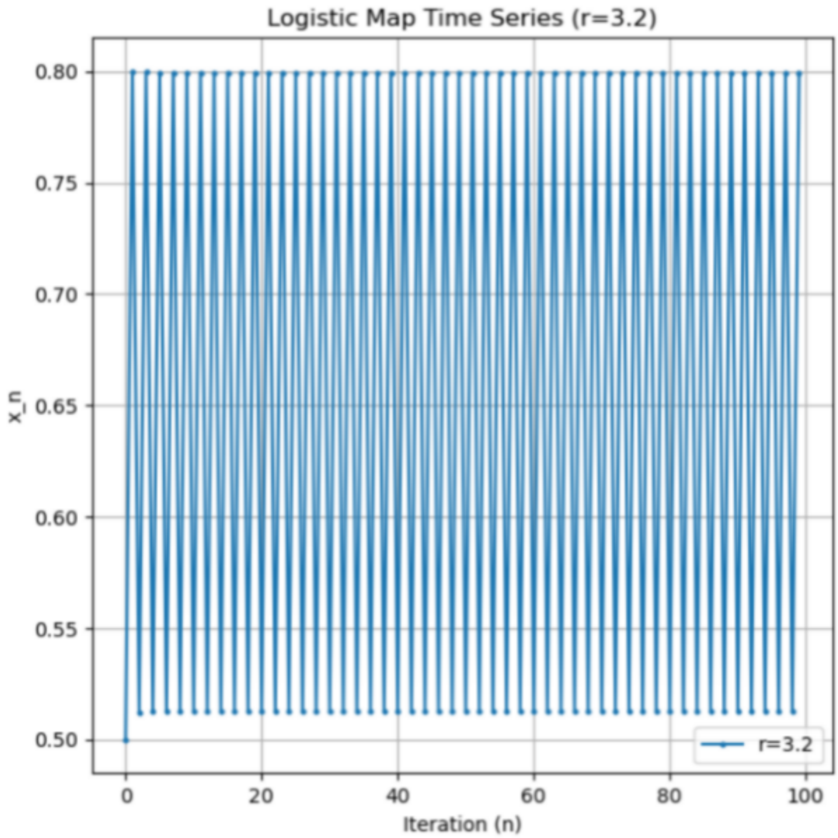


Figure 5: Time series graph when r=3.2

By simultaneous equations:

$$x_{n+2} = x_n \tag{3}$$

$$x_{n+2} = r\left(rx_n(1 - x_n)\right)\left(1 - rx_n(1 - x_n)\right) \tag{4}$$

Substituting $r = 3.2$ and solving for x_n, we get the specific values of x_n as 0.799 and 0.513.

In a alternative way, We can just use xn as the horizontal coordinate and x_{n+2} as the vertical coordinate to draw the curve in Desmos. The intersection of this curve and the straight line $x_{n+2} = x_n$ is its fixed point. By looking at the graph, we find that it has two intersection points with $x_{n+2} = x_n$, which are 0.799 and 0.513.

For different r values, we can use the above method to get all the r values in the interval. When we plot the obtained values with parameter r as the x-axis and the obtained x_n values as the y-axis, we can find that the image is bifurcated, which also explains why such a shape appears in the bifurcation graph of the logistic map.

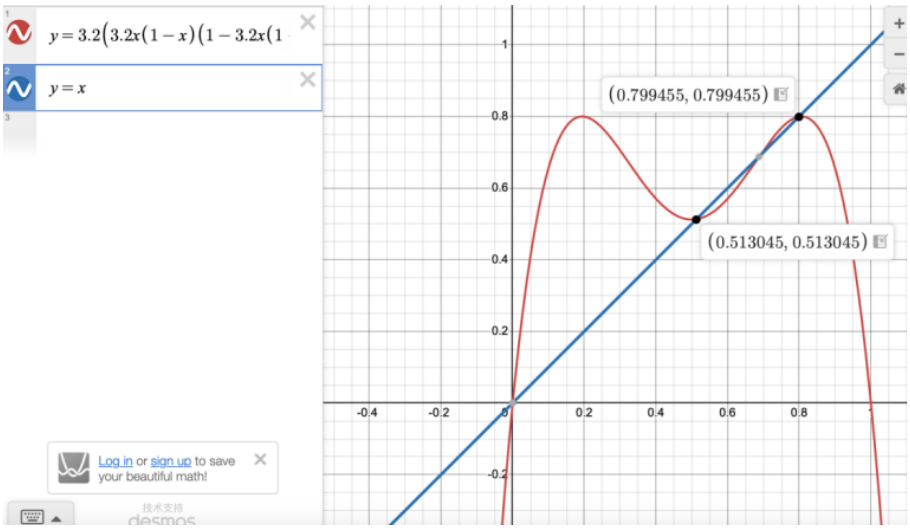


Figure 6: Intersection of $x_{n+2} = x_n$

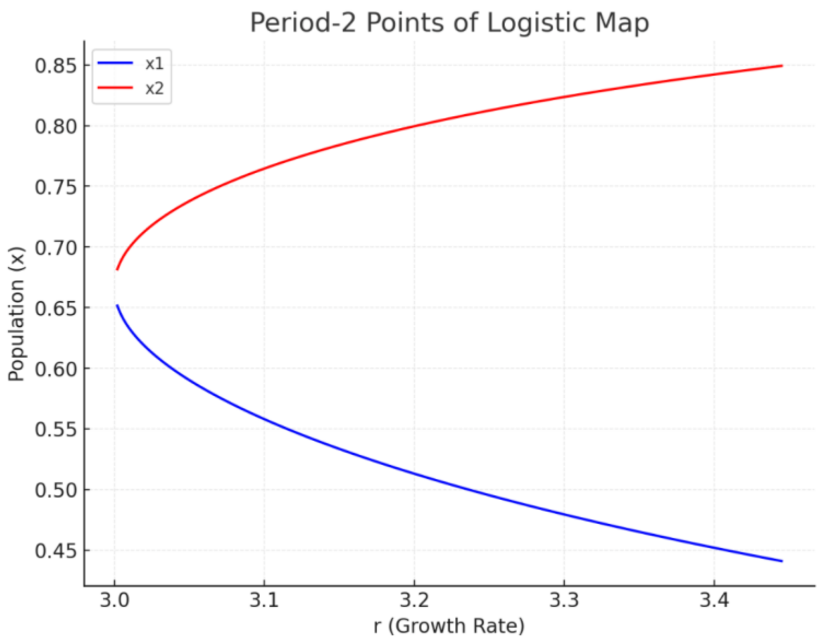


Figure 7: Relationship between x and parameter r

3.2 Feigenbaum Constant

Observing the bifurcation diagram, we find that as r gradually increases, the number of system iteration cycles will gradually increase, from 1 to 2, 2 to 4, etc. We record the parameter values r corresponding to these period bifurcation points as r_1, r_2, r_3, etc. As the period gradually doubles, we will find that the intervals between the parameter value sequences corresponding to the doubling period points will become smaller and smaller, and seem to show an exponential decay trend. We divide the distance between the first two bifurcations by the distance between the next two bifurcations to get the ratio limit:

$$\lim_{n \to +\infty} \frac{r_{n-1} - r_{n-2}}{r_n - r_{n-1}} = \delta \tag{5}$$

Then we substitute each period and the value of r when each period occurs into this formula for calculation, and obtain the following table.

n	Period	r_n	δ
1	2	3.0	-
2	4	3.4494897	-
3	8	3.5440903	4.7514
4	16	3.5644073	4.6562
5	32	3.5687594	4.6683
6	64	3.5696916	4.6686
7	128	3.5698913	4.6692
8	256	3.5699340	4.6694

Table 1: Bifurcation Points and Feigenbaum Constant

The table shows that the calculated results will get closer and closer to a constant: 4.6694, which is called the Feigenbaum constant. It is used to commemorate its discoverer, the American physicist Feigenbaum. Its exact value can be written as:

$$\delta = 4.66920160910299067185320382 \cdots \tag{6}$$

Feigenbaum's constant is not only valid for the Logistic mapping equation, it also has similar period doubling phenomenon and convergence speed for single peak mapping. This is not only a mathematical phenomenon, but also includes physical fluids, circuits, lasers and chemical reactions. Feigenbaum's constant has been found in these systems, and the errors obtained are all within the error range of 4.6692016. This shows the universality of Feigenbaum's constant. The universality of Feigenbaum's constant is applicable to nonlinear dynamic systems that meet the following conditions:

1. The system is a one-dimensional mapping (such as parabolic mapping, trigonometric function mapping), usually expressed as $x_{n+1} = f(x_n, r)$, where $f(x)$ is continuous. The meaning of one-dimensional mapping is to map a one-dimensional real number (or a real number interval) to itself.

2. The mapping has a single peak (such as the parabolic shape in the Logistic mapping). A single peak means that the function graph has a unique local maximum point on the domain of definition.

3. The behavior of the system undergoes period-doubling bifurcations as the control parameter r changes (from a fixed point to periods 2, 4, 8, etc., and finally into chaos)

4 Period-Three Behavior

4.1 Identification of the First Period-Three Cycle

The logistic map is defined as:

$$x_{n+1} = rx_n(1 - x_n).$$

Numerical and analytical studies of the logistic map show that the first occurrence of a period-three cycle is at $r \approx 3.82878$, the system exhibits a period-three behavior. This means there exist three distinct values x_1, x_2, x_3 such that:

$$x_1 \to x_2 \to x_3 \to x_1.$$

This marks a critical transition in the dynamics of the system.

4.2 Significance of Period-Three Orbits in the Context of the Yorke-Li Theorem

According to the Yorke-Li theorem, a one-dimensional map must have chaotic dynamics and orbits of all other periods if it has a period-three orbit. With $r \approx 3.82878$, this suggests:

- Coexistence of periodic orbits of all periods.
- Presence of chaos, where trajectories are highly sensitive to initial conditions.

The behavior in period three foreshadows chaos and emphasizes the intricate dynamics of the logistic map.

4.3 Cobweb Diagram

A *cobweb diagram* is a graphical representation used to visualize the iterative behavior of a one-dimensional map, such as the logistic map. It provides an intuitive way to understand how an initial value evolves over time through successive applications of the map. For the logistic map:

$$x_{n+1} = rx_n(1 - x_n),$$

the cobweb diagram illustrates the interaction between the function $y = rx(1 - x)$ and the line $y = x$, where values remain unchanged.

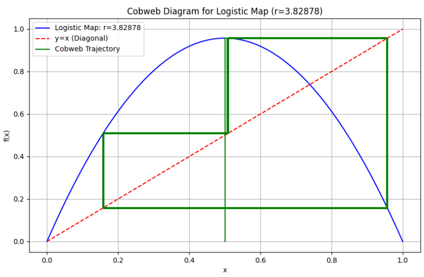


Figure 8: Cobweb Diagram for Logistic Map at $r = 3.82878$

For $r = 3.82878$, the cobweb diagram shows the system's dynamics in the *period-three cycle*:

1. **Period-Three Behavior**:
   - The diagram reveals that the trajectory alternates between three distinct points x_1, x_2, and x_3, forming a repeating loop:
$$x_1 \to x_2 \to x_3 \to x_1.$$
   - This behavior is characteristic of a stable period-three cycle in the logistic map.
2. **Triangular Repeating Pattern**:
   - The cobweb lines trace out a repeating triangular pattern in the diagram, visually demonstrating the periodic nature of the system.
3. **Stability of the Period-Three Cycle**:
   - The periodic orbit is stable for this value of r, as indicated by the consistent, repeating pattern.

5 Yorke-Li Theorem and implication on Logistic Map

The Yorke-Li theorem provides a crucial insight into the dynamical systems. It establishes that the existence of a period-3 orbit in a continuous map guarantees the presence of periodic orbits of all periods. This theorem not only highlights the deterministic nature of chaos but also bridges periodic behavior with unpredictability.

5.1 Proposition 10.4 (Li and Yorke):

Let f be a continuous function from $\mathbb{R}$ to itself. Assume that:

1. f has a period-3 point, or
2. There exists a point x_0 such that:
   - $f^3(x_0) \leq x_0 < f(x_0) < f^2(x_0)$, or

- $f^3(x_0) \geq x_0 > f(x_0) > f^2(x_0)$,

then f has periodic points of all periods $n \in \mathbb{N}$.

5.2 Implications on Logistic Map:

The theorem claims that the existence of a period-3 orbit in a dynamical system guarantees the emergence of periodic orbits of all periods and the chaotic dynamics. Once period-3 behavior is observed, the system transitions to a regime where periodicity coexists with chaos.

5.3 Sharkovskii's Theorem

Sharkovskii's theorem establishes a specific ordering of periodic points for a continuous function $f : \mathbb{R} \to \mathbb{R}$. The ordering of possible periods is given as:

$$3 \succ 5 \succ 7 \succ \cdots \succ 2{\cdot}3 \succ 2{\cdot}5 \succ 2{\cdot}7 \succ \cdots \succ 2^2{\cdot}3 \succ 2^2{\cdot}5 \succ \cdots \succ \cdots \succ 2^n \succ \cdots \succ 2^2 \succ 2 \succ 1.$$

The theorem claims that if f has a periodic point of period p, then f must also have periodic points of all periods q such that $q \prec p$ in this ordering. Specifically, if f has a period-3 point, then f has periodic points of all periods $n \in \mathbb{N}$, which lays the **groundwork for the Yorke-Li theorem**.

5.4 Evidence of Period-3 Behavior

5.4.1 Bifurcation Diagram

Figure 9 highlights the bifurcation diagram for $r > 3.57$, where the system transitions from periodic behavior to chaos. At $r \approx 3.83$, the diagram distinctly splits into three branches, indicating the onset of period-3 orbits. After the period-3 orbit shows up:

- n-period orbits appear, consistent with Yorke-Li's theorem.
- For higher values of r, chaotic behavior emerges, characterized by a dense structure in the diagram.

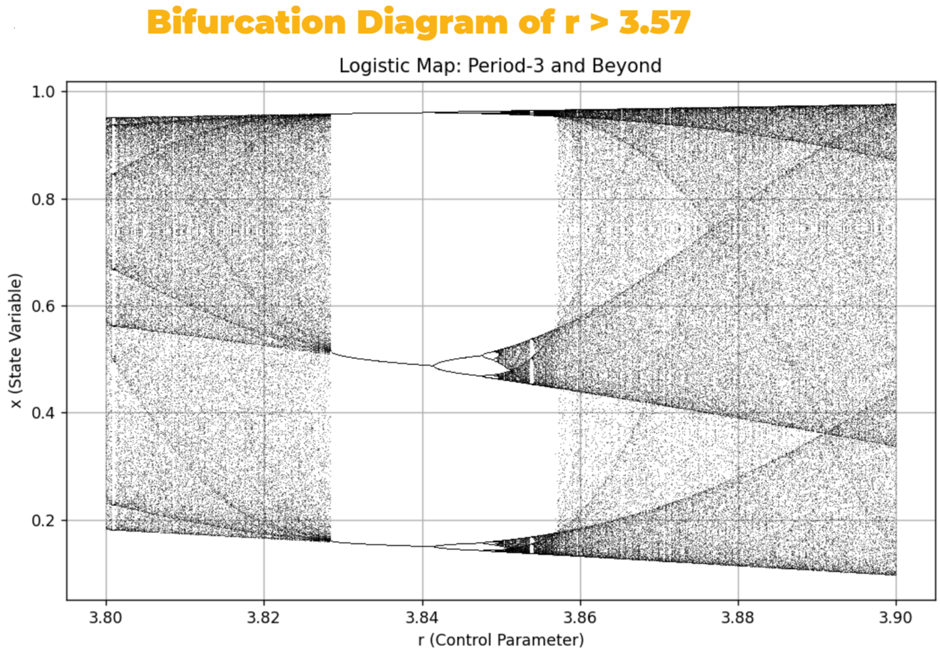


Figure 9: Bifurcation diagram for $r > 3.57$ showing the onset of period-3 window and chaotic behavior.

5.4.2 Cobweb Diagrams

Figure 10 shows the cobweb diagrams at three different values of r, illustrating the transition into chaos. At $r = 3.57$ (left panel), the system begins to show aperiodic trajectories that appear structured but no longer converge to a single fixed point or simple cycle. This marks the onset of chaotic behavior. At $r = 3.84$ (middle panel), the system exhibits clear period-3 behavior. The trajectory alternates among three distinct points, forming a closed cycle that repeats every three iterations, strongly supporting the existence of a period-3 orbit as predicted by Yorke-Li's theorem. Finally, at $r = 4.0$ (right panel), the system enters full chaos. The trajectories densely fill the space without repetition.

The presence of period-3 orbits guarantees, as the claim of Yorke-Li's theorem, the existence of n-period orbits for all $n \in \mathbb{N}$. This behavior leads to chaos.

5.5 Mechanisms of Chaos

Stretching and Folding: Chaos in the logistic map is driven by two key mechanisms:

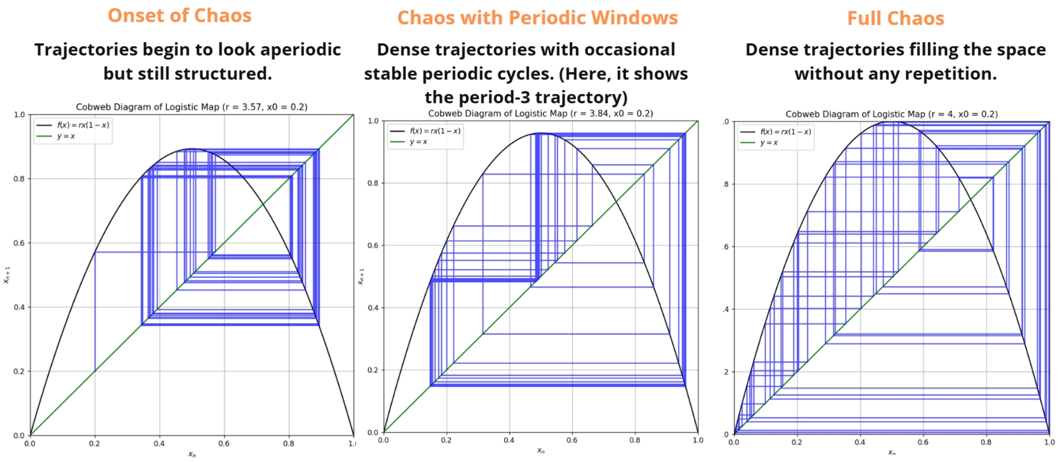


Figure 10: Cobweb Diagrams of the Logistic Map: Transition from the Onset of Chaos to Full Chaos.

- *Stretching:* Small differences in initial values of x are amplified as r increases, causing trajectories to diverge significantly over time. This explains the sensitive dependence on initial conditions observed in chaotic behavior.

- *Folding:* The quadratic nature of the logistic map ensures trajectories remain bounded within $[0, 1]$, while introducing overlap in the trajectories. This folding process creates the intricate structure seen in the bifurcation diagram and cobweb plots.

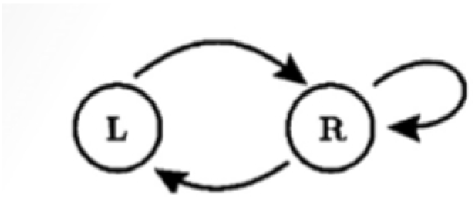
These mechanisms together explain the deterministic yet unpredictable dynamics of the logistic map, transitioning from periodic behavior to full chaos as r increases.

5.6 Intuitively explanation of the proof of Yorke-Li Theorem

The proof of the Yorke-Li theorem shows that a period-3 orbit ensures the existence of all periodic orbits ($n \in \mathbb{N}$) and chaotic behavior. The idea starts by dividing the domain into intervals I_L and I_R. Points in I_L map to I_R, while points in I_R map back to either I_L or I_R.

A transition graph (Figure 11a) represents these movements, and symbolic sequences (Table 11b) show periodic behaviors, like $(RL)^\infty$ for period-2 and $(RRL)^\infty$ for period-3. These sequences generalize to all n-period orbits.

The mechanism of chaos arises through stretching (amplifying small differences) and folding (keeping trajectories bounded), ensuring dense periodic points and sensitive dependence on initial conditions. Together, these steps connect the deterministic rules to chaotic dynamics, fulfilling the Yorke-Li theorem.



(a) Transition graph illustrating movement between I_L and I_R.

n	Symbol Sequence
1	R^∞
2	$(RL)^\infty$
3	$(RRL)^\infty$
n	$(R^{n-1}L)^\infty$

(b) Sequence of symbols for periods for a function with a period-3 point.

Figure 11: Yorke-Li proof components: transition graph and symbolic sequences.

6 Applications of Chaos Theory

6.1 Controlling Chaos

- **Yorke-Li Theorem**: Identifying periodic orbits enables the stabilization of chaotic systems, offering practical solutions:
  - *Medicine*: Stabilizing chaotic heart rhythms.
  - *Economics*: Managing market fluctuations by targeting chaotic dynamics.

6.2 Explaining Randomness in Deterministic Systems

- **Cryptography**: Using chaotic systems to generate secure pseudo-random sequences.
- **Weather Forecasting**: Demonstrating the sensitivity of chaotic systems to initial conditions and the inherent limits of long-term predictability.

6.3 Mandelbrot Set and Logistic Map

- The bifurcation diagram of the logistic map resembles cross-sections of the Mandelbrot set, showcasing the universal characteristics of chaotic systems.

6.4 Lorenz System and William's Theorem

- The **Lorenz attractor** represents a chaotic system with deterministic rules, characterized by sensitive dependence on initial conditions and aperiodic trajectories.
- **William's Theorem**: Confirms the chaotic attractor in the Lorenz system, validating its structure and properties as a chaotic system.

6.5 Other Chaotic Systems

- **Double Pendulum**: A tangible example of chaotic motion observed in physical systems. The double pendulum demonstrates how deterministic rules can give rise to highly complex and unpredictable trajectories, making it a classic illustration of chaos in real-world mechanics.

These systems, including the logistic map, highlight the common foundation of chaotic dynamics: deterministic rules governing behavior that appears unpredictable and complex.

Conclusion

The logistic map demonstrates how simple mathematical rules can lead to complex behaviors, from stable fixed points to chaotic dynamics. By analyzing fixed points and their stability, we observed how varying the parameter r drives the system through bifurcations, leading to period-doubling cascades and eventually chaos. The bifurcation diagram clearly shows these transitions, supported by the universal Feigenbaum constant.

The Yorke-Li theorem highlighted the importance of period-3 behavior, marking the point where the system guarantees periodic orbits of all periods and transitions to chaos.

These findings emphasize the logistic map's relevance to reality applications. Its dynamics mirror real-world chaotic systems like the Lorenz system, and have practical applications in cryptography, weather prediction, and understanding nonlinear systems. This study underlines how a simple model like the logistic map can reveal the underlying order and unpredictability in complex systems.

References

1. Robinson, R.C. (2012). *An Introduction to Dynamical Systems: Continuous and Discrete.* American Mathematical Society.

2. Feigenbaum, M.J. (1978). "Quantitative Universality for a Class of Nonlinear Transformations." *Journal of Statistical Physics.*

3. Mosekilde, E. (2009). *Period doubling. Scholarpedia, 4*(7), 6439. Retrieved from `http://www.scholarpedia.org/article/Period_doubling`.

4. Li, T.Y., Yorke, J.A. (1975). "Period three implies chaos." *The American Mathematical Monthly*, 82(10), 985–992.

5. Lorenz, E.N. (1963). "Deterministic nonperiodic flow." *Journal of the Atmospheric Sciences*, 20(2), 130–141.

6. Mandelbrot, B.B. (1982). *The fractal geometry of nature.* W. H. Freeman and Co.

7. Williams, R.F. (1979). "The structure of Lorenz attractors." *Publications Mathématiques de l'IHÉS*, 50, 73–99.

8. Strogatz, S.H. (2018). *Nonlinear dynamics and chaos: With applications to physics, biology, chemistry, and engineering.* CRC Press.

9. Gleick, J. (1987). *Chaos: Making a new science.* Viking.

Appendix: Python Code

This appendix contains the Python code used in this report.

Code for Logistic Map Iteration

```python
# Logistic Map Iteration
def logistic_map(r, x, n):
    """Perform n iterations of the logistic map."""
    results = []
    for i in range(n):
        x = r * x * (1 - x)
        results.append(x)
    return results

# Example usage
r = 3.7
x0 = 0.5
iterations = 100
trajectory = logistic_map(r, x0, iterations)
```

Code for Bifurcation Diagram

```python
import numpy as np
import matplotlib.pyplot as plt

r_values = np.linspace(2.5, 4.0, 10000)
iterations = 1000
last = 200
x = 0.5 * np.ones(len(r_values))

for i in range(iterations):
```

```python
        x = r_values * x * (1 - x)
        if i >= (iterations - last):
            plt.plot(r_values, x, ',k', alpha=0.1)

plt.title("Bifurcation Diagram of Logistic Map")
plt.xlabel("r")
plt.ylabel("x")
plt.show()
```

Code for Cobweb Plot and zooming out Bifurcation Diagram

```python
import numpy as np
import matplotlib.pyplot as plt


# Define the Logistic Map
def logistic_map(r, x):
    return r * x * (1 - x)


# Parameters
r_values = np.linspace(3.8, 3.9, 1000)  # Focus on the period-3
    region
n_iterations = 1000  # Total iterations
last = 200  # Only keep the last `last` iterations to show long-
    term behavior
x_initial = 0.5  # Starting value of x

# Prepare data for bifurcation diagram
x = np.ones(len(r_values)) * x_initial
bifurcation_x = []
```

```python
bifurcation_r = []

for i in range(n_iterations):
    x = logistic_map(r_values, x)
    if i >= (n_iterations - last):  # Store only the last values
        bifurcation_r.extend(r_values)
        bifurcation_x.extend(x)

# Plot the bifurcation diagram
plt.figure(figsize=(10, 6))
plt.plot(bifurcation_r, bifurcation_x, ',k', alpha=0.5,
   markersize=0.5)
plt.title("Logistic Map: Period-3 and Beyond")
plt.xlabel("r (Control Parameter)")
plt.ylabel("x (State Variable)")
plt.grid()
plt.show()

r = 3.83
x = 0.5  # Initial value
trajectory = []

for _ in range(100):
    x = logistic_map(r, x)
    trajectory.append(x)

plt.figure(figsize=(8, 4))
plt.plot(trajectory, '-o', markersize=4)
plt.title("Period-3 Trajectory at r = 3.83")
plt.xlabel("Iteration")
plt.ylabel("x")
```

```
plt.grid()
plt.show()

import numpy as np
import matplotlib.pyplot as plt



def logistic_map(x, r):
    return r * x * (1 - x)



def cobweb_plot(r, x0, n_iterations):
    # Set up the figure
    plt.figure(figsize=(8, 8))
    x = np.linspace(0, 1, 500)
    plt.plot(x, logistic_map(x, r), 'k',
             label='$f(x) = r x (1 - x)$')  # Logistic Map
    plt.plot(x, x, 'g', label='$y = x$')  # Identity line

    # Iterative process
    x_n = x0
    trajectory = [x_n]
    for _ in range(n_iterations):
        x_next = logistic_map(x_n, r)
        trajectory.append(x_next)
        x_n = x_next

    # Plotting the cobweb
    x_vals = [trajectory[0]]
    y_vals = [trajectory[0]]
    for i in range(1, len(trajectory)):
```

```python
        x_vals.extend([trajectory[i - 1], trajectory[i]])
        y_vals.extend([trajectory[i], trajectory[i]])
    plt.plot(x_vals, y_vals, 'b', alpha=0.7)

    # Formatting the plot
    plt.title(f'Cobweb Diagram of Logistic Map (r = {r}, x0 = {x0
       })')
    plt.xlabel('$x_n$')
    plt.ylabel('$x_{n+1}$')
    plt.legend()
    plt.grid(True)
    plt.xlim(0, 1)
    plt.ylim(0, 1)
    plt.show()


# Parameters for chaotic behavior
r = 4
x0 = 0.2
n_iterations = 50

for r in [3.52, 3.57, 3.84, 4]:
    cobweb_plot(r, x0=0.2, n_iterations=50)

import numpy as np
import matplotlib.pyplot as plt


# Logistic Map Function
def logistic_map(x, r):
    return r * x * (1 - x)
```

```python
# Bifurcation Diagram
r_values = np.linspace(3.82, 3.85, 5000)  # Zoomed range for
    period-3
iterations = 1000  # Total iterations
last = 200  # Only plot the last 200 iterations
x = 0.5 * np.ones(len(r_values))  # Initial condition for all r

# Iterating the Logistic Map
results = []
for i in range(iterations):
    x = logistic_map(x, r_values)
    if i >= (iterations - last):
        results.append(np.copy(x))

# Convert results to a 2D array
results = np.array(results)

# Plotting the Bifurcation Diagram
plt.figure(figsize=(10, 6))
plt.plot(r_values, results, ',k', alpha=0.25, markersize=0.5)
plt.title("Period-3 Window in the Logistic Map")
plt.xlabel("r (Control Parameter)")
plt.ylabel("x (State Variable)")
plt.grid(True)
plt.show()
```

Time series graph of Logistic Map

```python
import numpy as np
```

```python
import matplotlib.pyplot as plt

# Logistic map function: x_{n+1} = r * x_n * (1 - x_n)
def logistic_map(r, x0, n):
    x = np.zeros(n)
    x[0] = x0
    for i in range(1, n):
        x[i] = r * x[i - 1] * (1 - x[i - 1])
    return x

# Parameters
r_values = [2.5, 3.2, 3.5, 3.9]  # Growth rates
x0 = 0.5                         # Initial value
n = 100                          # Number of iterations

# Create individual plots for each r value
plt.figure(figsize=(12, 12))

for i, r in enumerate(r_values, 1):
    time_series = logistic_map(r, x0, n)

    # Create subplot
    plt.subplot(2, 2, i)
    plt.plot(range(n), time_series, linestyle="-", marker="o",
       markersize=2, label=f"r={r}")
    plt.title(f"Logistic Map Time Series (r={r})")
    plt.xlabel("Iteration (n)")
    plt.ylabel("x_n")
    plt.grid(True)
    plt.legend()
```

```python
# Adjust layout for better appearance
plt.tight_layout()
plt.show()
```

Code for Finding r Value

```python
import numpy as np

# Logistic map
def logistic_map(x, r):
    return r * x * (1 - x)

# Function to detect a period-three cycle for a given r
def detect_period_three(r, x0, max_iter=500):
    """
    Check if a given value of r produces a period-three cycle
       starting from x0.
    """
    x = x0
    trajectory = []
    for _ in range(max_iter):
        x = logistic_map(x, r)
        if len(trajectory) >= 3 and abs(x - trajectory[-3]) < 1e
           -6:
            return True  # Period-three detected
        trajectory.append(x)
    return False

# Find period-three onset
def find_period_three_onset(r_values, initial_guesses):
    """
```

```
    Search for the smallest r where a period-three cycle occurs.
    """
    for r in r_values:
        found_period_three = False
        for x0 in initial_guesses:
            if detect_period_three(r, x0):
                found_period_three = True
                break
        if found_period_three:
            return r
    return None

# Parameters for the search
r_values = np.linspace(3.8, 3.85, 1000 Number of samples to
   generate. Default is 50.)
# Search range for r
initial_guesses = np.linspace(0.1, 0.9, 10)  # Initial guesses
   for x0

# Find the onset of the period-three cycle
onset_r = find_period_three_onset(r_values, initial_guesses)

# Display the result
print(f"r is approximately {onset_r:.6f}")
```

Code for Cobweb Diagram

```
import numpy as np
import matplotlib.pyplot as plt
```

```
# Define the logistic map function
def logistic_map(r, x):
    return r * x * (1 - x)

# Parameters
r = 3.82878  # Parameter in the period-three window
iterations = 100  # Number of iterations
x_start = 0.5  # Initial condition for x

# Generate data for the cobweb diagram
x_vals = [x_start]
for _ in range(iterations):
    x_vals.append(logistic_map(r, x_vals[-1]))

# Prepare the plot
x = np.linspace(0, 1, 500)  # Range of x values
y = logistic_map(r, x)  # Logistic map curve

plt.figure(figsize=(10, 6))

# Plot the logistic map and the diagonal line
plt.plot(x, y, label=f"Logistic Map: r={r}", color="blue")
plt.plot(x, x, label="y=x (Diagonal)", color="red", linestyle="
   dashed")

# Plot cobweb trajectory
x_cobweb = [x_vals[0]]
y_cobweb = [0]  # Start at (x0, 0)
for i in range(1, len(x_vals)):
    # Vertical line to the curve
    x_cobweb.append(x_vals[i - 1])
```

```
        y_cobweb.append(x_vals[i])
        # Horizontal line to y=x
        x_cobweb.append(x_vals[i])
        y_cobweb.append(x_vals[i])

plt.plot(x_cobweb, y_cobweb, color="green", label="Cobweb
    Trajectory")

# Formatting the plot
plt.title(f"Cobweb Diagram for Logistic Map (r={r})")
plt.xlabel("x")
plt.ylabel("f(x)")
plt.legend()
plt.grid()

# Show the plot
plt.show()
```