Project 5: Discrete logistic growth

Suppose a perennial plant has a population $x(n)$ at year n, where x is measured in thousands. If the population were growing in an unbounded environment, the population obeys

$$x(n+1) = x(n) + r \cdot x(n) \tag{1}$$

where r is the per-capita growth rate. If instead the population is in a bounded environment, growth is limited, and the population obeys

$$x(n+1) = x(n) + r\left(1 - \frac{x(n)}{K}\right)x(n) \tag{2}$$

where K is a parameter we refer to as the carrying capacity.

1. Suppose $r = 0.1$ and $K = 0.6$. Sketch your intuition for the population $x(t)$ from a starting population $x(0) = 0.2$. What are the steady states? Which are stable, and which are unstable?

Write Matlab code to solve the dynamical system, and answer the following questions:

2. Suppose $r = 0.1$ and $K = 0.6$. Generate time series of the populations for a few starting populations $x(0)$. Does it match your intuition?
3. Suppose $r = 2.1$ and $K = 0.6$. Generate time series of the populations for a few starting populations $x(0)$.

In a discrete-time dynamical system, if the population cycles between two values, the solution is called a two-cycle. Cycling between N values is called an N-cycle.

4. Check that at $r = 2.5$ and $K = 0.6$ there is a 4-cycle. Can you find a 3-cycle?
5. In this part, we will do a parameter sweep for $0 < r < 3.0$, with fixed $K = 0.6$. The goal is to generate a diagram where the horizontal axis is the parameter value r. On the vertical axis, if there is a stable steady state, plot the steady-state population. If there is an N-cycle, plot the N values of x that it cycles through. [1]
    - Hint: One way to plot the steady state or the N-cycle is to simulate the system until $n_{\max}$, and plot the last half values of $x(n)$. You need to choose $n_{\max}$ large enough so that the dynamics have settled into their steady state (or steady cycle) by $n_{\max}/2$.
    - Hint: How many r values should you explore?

[1] This type of behavior in a dynamical system is called *chaos*! This particular type of chaos is called period-doubling chaos.