

Plot the effect of growth rate on a population following discrete logistic growth

Po-Ju Ke

Introduction

In class, we have built a function that simulates the growth of a population that follows discrete geometric growth. This is assuming that resources are unlimited and therefore population growth does not slow down as the population size continues to increase. If resources are limited, the population will instead follow so-called “discrete logistic growth”, which can be written as:

$$N_{t+1} = N_t \left[1 + r \left(1 - \frac{N_t}{K} \right) \right]$$

where r is the species’ intrinsic growth rate and K is the species’ carrying capacity (i.e., the maximum population size that the system can sustain). You can see that if $N_t = K$, N_{t+1} will be equal to N_t and the population stops growing. While intuitively we would expect that the population time series will approach the carrying capacity smoothly (panel a in the figure below), in reality, the time lag associated with discrete population growth may create many interesting dynamics. For example, if the intrinsic growth rate is high, the population may overshoot and reach the carrying capacity in a zig-zag motion (panel b). Moreover, if the intrinsic growth rate is too high, the population dynamics appear to jump randomly and never stabilizes (research has shown that it will demonstrate chaotic dynamics). In this question, you will re-create the figure below, which shows how the population time series varies depending on the species’ intrinsic growth rate.

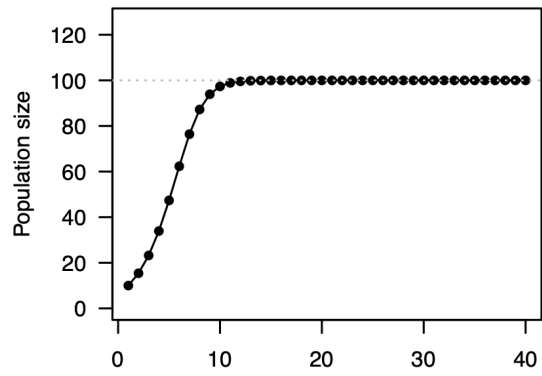
What to do

1. Create a function `Logistic`. This function will take in four arguments: (1) initial population size N_0 , (2) intrinsic growth rate, r , (3) carrying capacity, K , and (4) simulation time, T . The function will output a vector of size T , which begins at the initial population size and tells you the population size (following the discrete logistic growth equation above) at each time step.
2. Run the function three times and save the output into three different objects. All three simulations have the same initial population size of 10, carrying capacity of 100, and are all run for 40 time steps. The three simulations only differ in their intrinsic growth rate, with values equal to 0.6, 1.9, and 2.7, respectively.
3. Plot the figure, consisting of three panels, each showing the population time series (i.e., population size on the y-axis and time on the x-axis) under different values of intrinsic growth rate.
4. Make sure that you include the title of each plot (at the left upper corner of each panel), indicating the value of the intrinsic growth rate; also, the y-axis tick-mark labels are horizontal, and each panel has the same range of values on the y-axis (for this, you may need to find the maximum value of all simulated population time series). Also, include a dashed horizontal grey line passing through the carrying capacity ($K = 100$) to identify the expected population size if populations were to grow continuously.

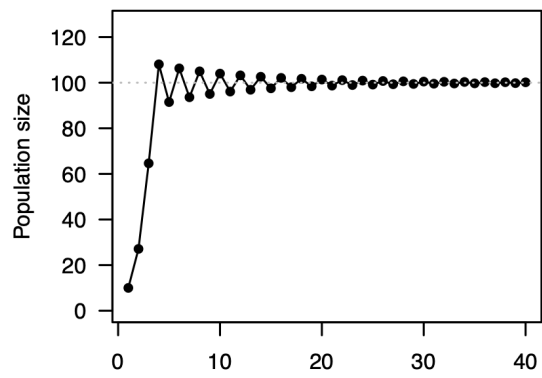
5. For the last panel, you will see that the population dynamics exhibit the so-called “chaotic motion”. After time step 5, We are worried if the population size drops below 50 and wish to highlight those points when such low population size occurs. Please add another red horizontal line indicating this threshold (i.e., 50) and highlight points that are below this threshold.
6. Save the figure into pdf file (height = 8 inches, width = 3 inches). The name of the file should be `DiscreteLogistic_XXXXXXXXXX.pdf`, where `XXXXXXXXXX` is your student ID.
7. Make sure that the R code is tidy and fully reproducible!

The figure should look like this (note that this is a single figure with three panels stacked together, not three figures):

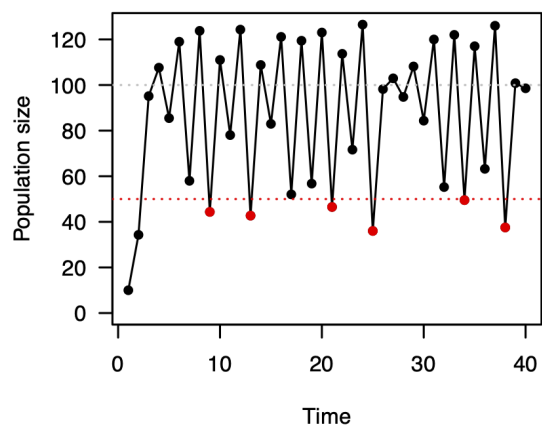
(a) $r = 0.6$



(b) $r = 1.9$



(c) $r = 2.7$



Hints

1. the function should have similar structure to the one you practiced in class. However, please note that you should return a `data.frame`, not a vector. This can be done by first creating the vector as our previous practice, combining it with a vector of `1:T` with the function `data.frame`, and then `return` it.
2. to initiate the vector containing population sizes, you can use `rep` or `vector` and manually modify the first element.
3. you will need the function `par` with argument `mfrow`.
4. you will need the high-level function `plot` with arguments such as, but not limited to, `las` and `ylim`.
5. for overlapping points and lines in `plot`, use `type = o`.
6. the easiest way to create the last panel is to first plot the whole thing in black, then add the red points over the original black lines/points.
7. highlight red points via subsetting and note that there are two criteria: time and population size.
8. you might need the low-level function `abline` with the argument `h` (horizontal) set to a specific value.
9. you might need the low-level function `title` with the argument `adj`.