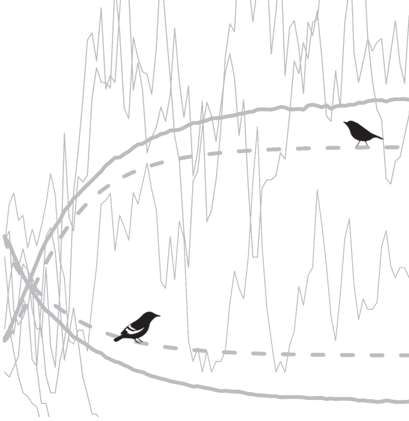
Contents



CHAPTER 1

Simple discrete-time models of single populations

Fibonacci's rabbit model

In the year 1202, the Italian mathematician Fibonacci considered the following problem:

> *A certain man had one pair of rabbits together in a certain enclosed place. One wishes to know how many are created from the pair in one year when it is the nature of them in a single month to bear another pair and in the second month those born to bear also.*

Fibonacci's rabbit problem may be the first published example of what we would now call a problem in theoretical ecology. Ecology seeks to explain the interactions among organisms and between organisms and their environment. Theoretical ecologists use verbal, conceptual, graphical, mathematical, and computational models of ecological systems to explore the logical consequences of assumptions and to make testable predictions that can be compared against data from field or laboratory studies. Before we go further, let us define the terms **theory** and **model**, as well as the related term **hypothesis**:

- A theory typically constitutes a broad and general set of ideas that are independent of any particular model. For example, we may have a theory of rabbit population dynamics that describes how growth rates respond to food availability, predator abundance, and so on.
- A model falls under the umbrella of a theory and makes more specific and more detailed assumptions. Models can be conceptual or verbal, but in this book we focus mainly on quantitative models, which can generate not only qualitative but quantitative predictions. A single theory may encapsulate many models. For example, later in the book we will meet the Lotka–Volterra competition model, which falls under the general umbrella of niche theory and makes very specific assumptions about how niches operate, thus facilitating quantitative modelling. We will primarily be concerned with mathematical models, but we will often motivate these with verbal, graphical, and conceptual models, and we implement many of them as computational models. Fibonacci's statement of his rabbits problem is a verbal model that he then turned into a mathematical model, as we will see below.

- The term "hypothesis" is sometimes used interchangeably with "theory," but typically a hypothesis has a more limited focus or less empirical support than a theory. For example, we may hypothesise that a rabbit population will grow without bound if there is no constraint on food availability.

There are grey areas between these terms and they are not always used as strictly defined above, but, for the purposes of clarity, in this book we will endeavour to stick to these definitions.

Fibonacci's statement of his rabbits problem is a verbal model that makes some of its assumptions explicit: the initial condition is one pair of rabbits; a pair of rabbits produces a new pair of offspring every month; and the time frame of interest is one year. But the verbal model is unsatisfactory, because other assumptions remain unstated: How long do rabbits live? When do new rabbits reach reproductive age? Is there some carrying capacity imposed by the environment? If we want to tackle the rabbits problem rigorously, we have to formulate a mathematical model. Mathematical modelling is really just a way of forcing ourselves to make all of our assumptions explicit and of exploring the logical consequences of these assumptions in a rigorous and unambiguous way. We can draw inspiration from the sentiments of Ronald Ross, a pioneering epidemiologist of the early 1900s who was working on disease ecology and emphasised the need to treat the subject mathematically, because "to say that a disease depends upon certain factors is not to say much, until we can also form an estimate as to how largely each factor influences the whole result. And the mathematical method of treatment is really nothing but the application of careful reasoning to the problems at issue." Ross's sentiment applies universally: treating a problem mathematically or computationally is just a way of forcing ourselves to be rigorous and logical.

Fibonacci did formulate his rabbit population mathematically and in so doing made the following more explicit assumptions: the rabbits can begin to reproduce in their second month of life; and the rabbits never die. He did not assume any kind of environmental carrying capacity, so the rabbit population can keep growing indefinitely. In modern notation, his mathematical model can be written as

$$F_{t+1} = F_t + F_{t-1} \tag{1.1}$$

where F_t is the number of pairs of rabbits at time t. In words, Eq. (1.1) says that the number of pairs of rabbits in month $t+1$ is equal to the number alive in the previous month t (because all rabbits survive) plus a number of offspring equal to the number alive in month $t-1$ (because only rabbits at least one month old can breed). Assume that prior to the start of the year there are no rabbits ($F_0 = 0$) but then a pair of newborn rabbits is introduced in month 1 ($F_1 = 1$). Then according to Eq. (1.1) we will have $F_2 = F_1 + F_0 = 1$, which biologically means that the pair survives to month 2 but can't yet breed. In month 3, Eq. (1.1) then gives us $F_3 = F_2 + F_1 = 2$, because now the original pair has produced one pair of young, and so on. We can easily iterate Eq. (1.1) by hand to compute the first several values of F_t for $t \geq 1$ as $\{1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144 \ldots\}$. The last value here gives us the answer to Fibonacci's problem: in month 12 there will be $F_{12} = 144$ rabbits. For higher values of t, we can easily compute F_t with a simple computer program that iterates Eq. (1.1) for us (Box 1.1). A time series of the Fibonacci model is shown in Fig. 1.1.

Box 1.1. R code for the Fibonacci rabbit model

To iterate the Fibonacci model (Eq. (1.1)) in R, we can use the following code:

```
1  # number of timesteps (months) over which to iterate model
2  t_max = 12
3
4  # initialise an empty vector to store the population sizes
5  # over time; the first value will correspond to month 0,
6  # i.e., t=0
7  Fs = numeric(t_max+1)
8  # set the initial population size to zero
9  Fs[1] = 0
10 # introduce a pair of rabbits in month t=1
11 Fs[2] = 1
12
13 # execute a loop from month t=2 to t=t_max, in each step
14 # calculating the value of the population size at month t
15 # from the Fibonacci formula
16 for ( t in 2:t_max )
17   Fs[t+1] = Fs[t] + Fs[t-1]
18
19 # plot the results
20 plot(0:t_max,Fs,pch=19,type='o',xlab='t',ylab='F')
```

Lines 1–11 set up the initial conditions and the object `Fs` for storing the population sizes. Lines 16–17 perform the work of iterating the model with a loop. Line 20 graphs the results. In line 20 the `pch` argument is set to `19` to draw the time series as solid points, and `type` is set to `'o'` to draw both ("o" for "overplotted") points and lines. A sample output graph is shown in Fig. 1.1. Longer time series can be generated by increasing the `t_max` parameter on line 2.

Be careful with indexing when converting a mathematical model to computer code. In mathematics (and some computer programming languages such as C++), indexing often starts at zero, but in R indexing starts at one, and thus line 9 sets `Fs[1]=0`, which corresponds to the mathematical statement $F_0 = 0$ (see text). For the same reason, we plot `0:t_max` on the horizontal axis instead of `1:(t_max+1)`.

A notational convention in code samples throughout this book is to use the character `s` at the end of an object name to indicate that the object is a vector comprising many individual values. Hence the object `Fs` comprises multiple values of `F`, with the `s` intended to evoke the English plural.

A final note: We recommend that any good R program start with the command `rm(list=ls())` to remove any existing objects in memory that may affect program execution, but for brevity we omit this command in the scripts presented in this book.

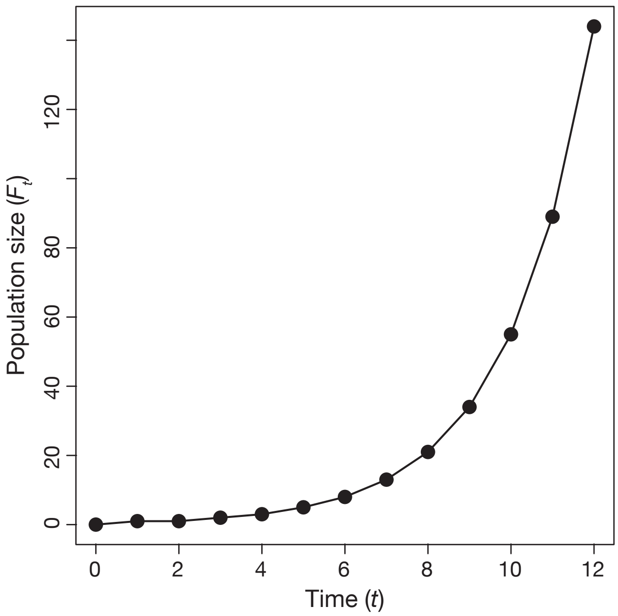


Fig. 1.1. Time series for the Fibonacci rabbit model (Eq. (1.1)), showing the number of pairs of rabbits F_t in each month t over one year (see code in Box 1.1).

Eq. (1.1) is an example of a **difference equation**. A difference equation relates the value of a variable at time $t+1$ to the value of the variable in previous timesteps. In this case the variable of interest is the rabbit population size, F. A defining characteristic of a difference equation is that the timesteps are discrete, so difference equations are **discrete-time models**. Eq. (1.1) is also an example of a **dynamical system**, a rule that describes how some variable changes over time. And it is a **one-dimensional system** because there is only one time-dependent variable and correspondingly only one dynamical equation. A two-dimensional system, for example, would require two equations. In this chapter we stick to one-dimensional systems, but in later chapters we move on to higher-dimensional systems.

We will leave the Fibonacci rabbit model for now. The Fibonacci model, our first ecological model in this book, was also arguably the first ecological model in history. It is very simple and perhaps unlikely to be applicable to any real population. Despite this, we have already learnt from it some basic modelling concepts and gleaned at least one biological insight, albeit one that is obvious to most ecologists: in the absence of an environmental carrying capacity, a population will tend to grow without bound. The Fibonacci numbers (i.e., the sequence of numbers generated by Eq. (1.1)) are important in mathematics and have found various applications in biology, beyond population models. We will revisit the Fibonacci rabbit model in Chapter 3. Next let us explore an even simpler model.

Geometric-growth model

Consider a population that grows at a constant rate each year so that the population in one year is double that in the previous year. We can express this mathematically as

$$N_{t+1} = 2N_t \tag{1.2}$$

This model is simpler than the Fibonacci rabbit model in the sense that the population now depends only on the population size in the previous timestep, not on the population size two timesteps ago. For a given initial population size N_0, the model can be projected forward in time by iterating Eq. (1.2) either by hand or on a computer. For example, if $N_0 = 20$, our first few values of N_t are {20, 40, 80, 160, 320, 640, . . .}.

We can generalise the model specified by Eq. (1.2) by allowing arbitrary fixed per capita birth and death rates, denoted by b and d, respectively. This leads to the equation

$$N_{t+1} = N_t + bN_t - dN_t$$

which can be rewritten as

$$N_{t+1} = (1 + b - d)N_t \tag{1.3}$$

For this model to make biological sense, the birth and death rates must be nonnegative ($b \geq 0$, $d \geq 0$), and the maximum value of the death rate must be $d = 1$, which would imply that all extant individuals die every timestep (as in, say, a model of annual plant population dynamics).

Because $b - d$ is just a fixed value, we can define $r \equiv b - d$ and write Eq. (1.3) as

$$N_{t+1} = (1 + r)N_t \tag{1.4}$$

We refer to the quantity r as the per capita population growth rate (it is sometimes called the **discrete growth factor**, and the factor $1 + r$ is called the **finite rate of increase**). The restrictions on b and d given above imply that $r \geq -1$. If $r > 0$ the population will grow over time, whereas if $r < 0$ the population will shrink over time (Fig. 1.2; see also code in Box 1.2). We refer to quantities that do not change over time, such as r, b, and d, as **parameters**. Let us clearly define what we mean by **variables**, **parameters**, and **constants**.

- A variable is a quantity that changes over time, e.g., population size N. Time (t) is itself a variable in the geometric-growth model and in most of the models we will consider. We can refer to variables in our models other than time as time-dependent variables.
- A parameter is a quantity that does not vary over time and that is not given a numerical value in the model specification, e.g., r in Eq. (1.4).
- By "constant" we usually mean a numerical constant, such as 2 or π or e. Note that sometimes parameters are also referred to as "constants," because they are constant over time. Ambiguity can be avoided by using the terms "parameter" and "numerical constant."

Note that the time variable t in the geometric-growth model does not have units, because the timescale arises naturally here from the discrete mathematical treatment of time. Similarly, the parameters b and d, which refer to the rates at which individuals are born and die per timestep, do not have units either.

Let us get back to analysing the model. As shown in Boxes 1.1 and 1.2, although it is straightforward to iterate a difference equation such as

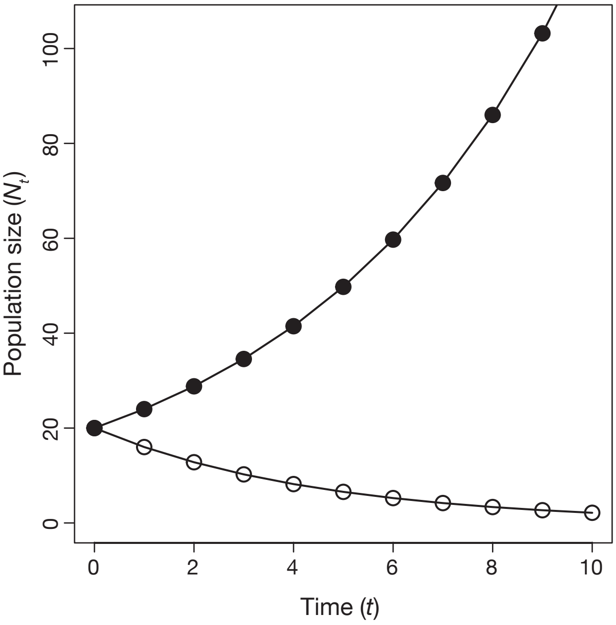


Fig. 1.2. Time series for the geometric-growth model (Eq. (1.4)) for initial population size $N_0 = 20$ and two values of the growth rate: $r = 0.2$ (solid points) and $r = -0.2$ (open points). Each time series shows the number of individuals N_t at the values of time t indicated (see code in Box 1.2).

Box 1.2. R code for the geometric-growth model

To iterate the geometric-growth model (Eq. (1.4)) in R, we can use the following code:

```
 1  # number of timesteps to iterate the model
 2  t_max = 10
 3
 4  # initial population size
 5  N0 = 20
 6
 7  # population growth rate
 8  r = 0.2
 9
10  # initialise an empty vector to store the population sizes
11  # over time
12  Ns = numeric(t_max+1)
13  # set the population size in the first timestep to N0
14  Ns[1] = N0
15
16  # execute a loop from time t=1 to t=t_max, iterating the map
17  for ( t in 1:t_max )
18    Ns[t+1] = (1+r)*Ns[t]
19
20  # plot the results
21  plot(0:t_max,Ns,pch=19,type='o',xlab='t',ylab='N',
22        ylim=c(0,max(Ns)))
```

The structure of the program is similar to that in Box 1.1, but now we must define a value for the growth-rate parameter `r` (line 8). In line 22, the `ylim` argument to the `plot()` function is used to expand the range of the y-axis to encompass $y = 0$. This is done for aesthetic purposes (it is often desirable to see a graph's y-axis start at zero). You can generate Fig. 1.2 by running the code above, which produces a time series for the growth rate $r = 0.2$, and then the code below, which produces a time series for a growth rate $r = -0.2$ and plots it on the same set of axes:

```
23  r = -0.2
24  for ( t in 1:t_max )
25    Ns[t+1] = (1+r)*Ns[t]
26  points(0:t_max,Ns,pch=1,type='o')
```

The `points()` command here tells R to use the same set of axes as for the previous graph, instead of drawing a new figure, and `pch=1` tells it to use open points, so we can visually distinguish this time series from the first one. The option `type='o'` again tells R to draw both points and lines, which is possible (perhaps counterintuitively) even though the command used is `points()` (we could instead use `lines()` and the output would be exactly the same).

Eq. (1.4) using a computer, this can be time consuming for large t or for complex models. For some models it is possible to obtain a mathematical solution, also called an **analytical solution**, that allows us to quickly compute the values of our variables at some arbitrary time t without computing all the intermediate values. The geometric-growth model (Eq. (1.4)) has such an analytical solution for N_t as an explicit function of time t and the model parameters:

$$N_t = (1+r)^t N_0 \tag{1.5}$$

This solution can be verified by plugging it back into Eq. (1.4) (Box 1.3).

Not all models have analytical solutions. The more complex a model is, the less likely it is to admit an analytical solution. As it turns out, the Fibonacci rabbit model (Eq. (1.1)) also has an analytical solution, but the derivation is somewhat more complicated and we leave it to Chapter 3. Whenever we can find analytical solutions they are very useful because

- they are fast to compute numerically;
- they reveal insights about the relationships between parameters and variables that are hard to glean from numerical simulations; and
- they allow us to draw general conclusions about model behaviour that apply over all possible values of the parameters, rather than just for specific numerical choices of parameter values.

For example, from Eq. (1.5), with the help of a computer, we can quickly compute the value of N_{100} for given values of r and N_0. Computing N_{100} iteratively using the code in Box 1.2 would be comparatively slow (though still lightning fast on a modern computer). For the Fibonacci rabbit model, the analytical solution in addition gives us insights about the long-term growth rate of the population that would be difficult to obtain from numerical iteration (see Chapter 3).

Box 1.3. Analytical solution of the geometric-growth model

The geometric-growth model is specified by the following difference equation (Eq. (1.4)):

$$N_{t+1} = (1+r)N_t \tag{B1.1}$$

To verify that a solution is (Eq. (1.5)):

$$N_t = (1+r)^t N_0 \tag{B1.2}$$

we plug Eq. (B1.2) into both the left- and right-hand sides of Eq. (B1.1):

$$(1+r)^{t+1} N_0 = (1+r)(1+r)^t N_0$$

which is true by the laws of indices, verifying the solution as required.

This solution (Eq. (B1.2)) is a key result that we use throughout the book as part of our solutions to more complicated models.

What else can we learn from the geometric-growth model? In general, we are interested in computing **equilibria** of our models. An equilibrium (sometimes also called a **fixed point**) is a state where the system will remain at rest, i.e., the system state does not change over time. Biologically, we care about equilibria because an undisturbed system may tend to an equilibrium after a sufficiently long period of time. The mathematical way of saying "the system state does not change over time" in a discrete-time model is $N_{t+1}=N_t$. Any value of N that satisfies this equation is an equilibrium, and we can denote it by a special symbol, typically N^*, $\tilde{N}$, or $\overline{N}$. In this book, we use the latter notation, $\overline{N}$, but readers should be careful not to confuse this with the use of the same notation in statistics to denote the mean of a variable.

In the geometric-growth model (Eq. (1.4)), for an equilibrium we require

$$\overline{N}=(1+r)\overline{N} \tag{1.6}$$

Let us assume that $r\neq 0$ (the case $r=0$ could be of interest to mathematicians, but in biology no growth rate is precisely equal to 0 and such borderline cases can usually be safely ignored). In this case, the only solution to the equation is $\overline{N}=0$, i.e., the only equilibrium is to have zero individuals in our population. This makes intuitive sense. The model describes a closed population, and if there are zero individuals now, there will be zero individuals in the future; i.e., the population size will not be changing over time and we will have an equilibrium. But if there is a nonzero number of individuals, then the population will be either growing ($r>0$) or shrinking ($r<0$); i.e., we will not be at equilibrium. (Note that for the Fibonacci rabbit model defined by Eq. (1.1), there is an equilibrium at $\overline{F}=0$.)

The next question we might ask of our model is whether the equilibrium we just found is stable. A **stable equilibrium** is one towards which the system will return after a perturbation from the equilibrium; an **unstable equilibrium** is one from which the system will move away after a perturbation. The concepts of stable and unstable equilibria are illustrated in Fig. 1.3. We can visualise a stable equilibrium as the bottom of a rounded cup in which we place a ball (Fig. 1.3a). If the ball is given a small perturbation, it will return to the bottom. We can visualise an unstable equilibrium as the top of an inverted cup (Fig. 1.3b). If a ball is placed in this position and then given a small perturbation, it will roll away, never to return.

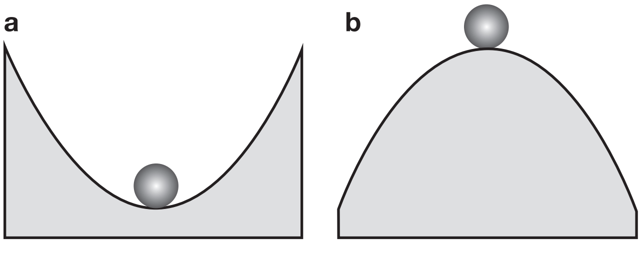


Fig. 1.3. A visual representation of the concepts of stable and unstable equilibria. (a) A stable equilibrium: The ball will remain in its current state if unperturbed, and if perturbed it will return to the original state. (b) An unstable equilibrium: Again the ball will remain in its current state if unperturbed, but now if perturbed it will move further away from the original state.

For the geometric-growth model, we can see intuitively that if the system is at the zero equilibrium ($\overline{N}=0$) and we perturb it by adding some individuals to the system, the system will return towards the zero equilibrium if and only if the growth rate r is less than 0. This is because if $r<0$ the population is decreasing at each timestep. On the other hand, if $r>0$ the zero equilibrium will be unstable. These properties can also be expressed in terms of the **eigenvalue** of the system at equilibrium, which for the geometric-growth model is $1+r$ (for a single-variable discrete-time model, such as Eq. (1.4), the eigenvalue is obtained by differentiating the right-hand side of the difference equation with respect to N_t and then plugging in the equilibrium value $N_t = \bar{N}$). Eigenvalues determine the growth rates of variables over time near an equilibrium. For a single-variable discrete-time system an equilibrium is stable if the system's eigenvalue at the equilibrium has magnitude less than one and unstable if it has magnitude greater than one, consistent with our intuitive deductions above about how r governs stability. Later in the book we will learn more about eigenvalues, including why they govern stability of equilibria and how to compute them for more general classes of models.

We have described the model specified by Eq. (1.4) as the geometric-growth model, but another name for it is the discrete-time exponential-growth model. We can see how the term "exponential" is relevant by writing $k = \log(1+r)$, such that the analytical solution to the model (Eq. (1.5)) becomes

$$N_t = e^{kt} N_0 \tag{1.7}$$

where e is the base of the natural logarithm. Note that e is a numerical constant (2.71828 . . .), and in this book we use $\log(x)$ to represent the natural logarithm of x, as is standard in ecology. Our trick of switching from a model with parameter r to a model with parameter k here is an exercise in **reparameterisation**. Reparameterisation is a handy tool for expressing our model in a different mathematical way and potentially getting new insights. We use reparameterisation repeatedly in this book.

Applications of the geometric-growth model

The geometric-growth model (Eq. (1.4)) makes many simplifying assumptions that are uncharacteristic of most real species populations. These assumptions include the following:

1. Constant per capita growth rate. The per capita growth rate is defined as $(N_{t+1} - N_t)/N_t$, which for this model is equal to r, a fixed value that is independent of population size. In reality, we might expect that the per capita growth rate changes with population size. In particular, it might become lower as the population grows and the environment becomes saturated.
2. Deterministic behaviour. There is no stochasticity (randomness) in this model. In reality, most populations are subject to stochasticity of various kinds.
3. No age structure. In most real populations, individuals of different ages have different vital rates (birth, death, and growth rates), and this affects overall population growth. In contrast, in the geometric-growth model all individuals are assumed to have the same vital rates regardless of age.
4. Population size treated as a continuous quantity. For $r = 0.2$ and $N_0 = 20$ in this model we obtain, for example, a population size of $N = 28.8$ at time $t = 2$. Obviously, in reality we cannot have 28.8 individuals. For small populations, the treatment of N as a continuous variable can be particularly problematic, as we will see in later chapters. (For now, when interpreting a model such as Eq. (1.4), the reader may assume that N measures the

population size in thousands or some other large multiplier, in which case the effects of biologically unrealistic fractional individuals turn out not to be too important.)

5. No sexual reproduction. The geometric-growth model does not define how new individuals are actually produced, and in particular there is no notion of sexual reproduction. Many real populations reproduce sexually, which can affect growth rates, especially when there is an uneven sex ratio.
6. No spatial structure. Real populations have spatial structure and this can affect their growth rates too. For example, the population growth rate may be locally higher where the population density is lower and the environment is less saturated.
7. No influences from other species. Most real populations interact with other species, for example, via competitive, mutualistic, and predator–prey interactions.
8. Closed population. This assumption may be realistic in some cases, but many populations are affected by immigration and emigration.
9. Population growth in discrete timesteps. This assumption is suitable for populations with non-overlapping generations or a fixed breeding schedule. But it may not be appropriate for populations that breed continuously over time.

We might wonder, given that no real population satisfies all these assumptions, whether the geometric-growth model is actually useful for anything. The answer is yes: the model can be useful in situations where the violations of the assumptions, in particular the first assumption that the per capita growth rate is constant, are not too severe.

For instance, imagine a laboratory experiment in which a population of bacteria is initialised with just a small number of cells and left to grow in a medium. For several hours this population may exhibit close to geometric growth, and if we had an estimate of the growth rate r, we could start the experiment and then use Eq. (1.5) to estimate how long it would take to reach some target population size, which could correspond to the bacterial density required for the next phase of the experiment. To do this, we would need to invert Eq. (1.5) to express t as a function of N_0:

$$t = \frac{\log\left(\dfrac{N}{N_0}\right)}{\log(1+r)} \tag{1.8}$$

Imagine that we start with a population of $N_0 = 10$ cells, that the target population size is $N_0 = 10^4$ cells, and that we have previously estimated the growth rate to be $r = 0.1$ per hour. We can then set our timestep length to be one hour and estimate from Eq. (1.8) that it will take about $t = 72$ hours to reach the target. We can then happily go home from the lab after lunch on Friday and enjoy the weekend, knowing that our bacteria will be ready for the next phase of the experiment by Monday afternoon. The key to the geometric-growth model being potentially useful here is that the population starts far below carrying capacity, so its per capita growth rate may be nearly independent of population size for some time, consistent with assumption 1 above. In addition, most of the other assumptions may be close to satisfied (e.g., there is asexual reproduction and the population is closed).

The simple geometric-growth model can also be applicable to problems involving invasive species or species recovering from very low numbers. For example, the sea otter

(*Enhydra lutris*) was extensively hunted for its fur in the nineteenth century, and declined precipitously in California to a low of about 50 individuals in the early twentieth century. After the species was protected, it recovered and underwent a period of almost geometric growth, which is well described by Eq. (1.5) (Fig. 1.4).

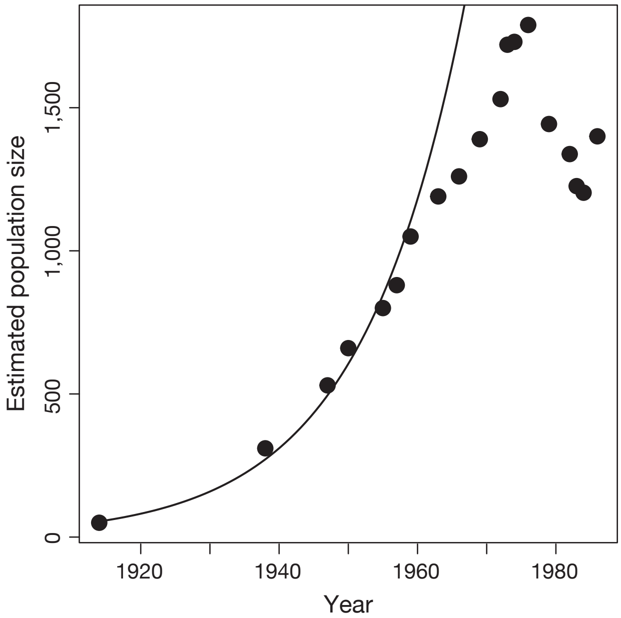


Fig. 1.4. Populations starting from low densities can exhibit growth that is close to geometric for a period of time. Points show estimated population sizes of sea otters in California over the period 1914–1986. The curve shows a fit of the analytical solution of the geometric-growth model (Eq. (1.5)) to the first seven data points up to the year 1959. After this point, population growth slowed down. (Data from Lubina & Levin 1988.)

Another general application of geometric growth is to systems where the population size starts small and the per capita growth rate is negative ($r < 0$). In such a scenario, the model's behaviour at high values of population size is irrelevant, and thus the absence of a carrying capacity is not a major limitation. For example, if we have an endangered species that is rare and declining, a geometric-growth model may yield a meaningful estimate of the species' expected time to extinction.

In many real scenarios, however, we are interested in populations that are close to their carrying capacity or may become so within the time frame of interest (see Chapter 2). In addition, violations of the other assumptions listed above may in some cases render the geometric-growth model inappropriate for projecting a population into the future.

But a general lesson here is that models can be useful even when their assumptions are not perfectly satisfied in nature. Indeed no model's assumptions will ever be perfectly satisfied. An imperfect model (i.e., every model) can be useful in two ways. Firstly, if the assumptions are not too severely violated, the model may be useful for making predictions and projections (e.g., Fig. 1.4). Alternatively, if a model's assumptions are severely violated, the inconsistency between the model's predictions and data can inform us of this fact and point to important mechanisms that are operating in nature and missing from our model. For example, when we see the sea otter population trajectory in Fig. 1.4 dropping below the geometric trend in later decades, we can infer that some process is operating that is not encapsulated in the geometric-growth model, for instance, food limitation.

Why and how do we model ecological systems?

The insights from the previous section are the basis for statistician George Box's well-worn aphorism that "All models are wrong but some are useful." A more nuanced version of this statement might be that "All models make unrealistic assumptions but some can nevertheless be useful." Before building a model, any ecologist, whether a novice student or a seasoned researcher, would do well to ponder the motivation for model building. What makes a model useful?

There are, broadly speaking, two uses of ecological models. The first is to make quantitative predictions. Predictions can be useful for practical applications, e.g., for predicting the spread of an invading or reinvading species (e.g., Fig. 1.4). Predictions can also be useful for testing the validity of ecological theories, as we will see at several points in this book. The second broad use of models is to give general insights into how ecological systems work. For example, the geometric-growth model allows us to attain a minimalist but very general understanding of population dynamics in the absence of a carrying capacity, stochasticity, or other more complicating factors. In 1798 Thomas Malthus used a geometric model of human population growth to predict a future demographic catastrophe. This catastrophe has at the time of writing not materialised, but the insights from Malthus's simple model did inspire both Charles Darwin and Alfred Wallace while they were developing their ideas about evolution by natural selection.

After we know why we want to build a model, how do we actually go about model building? Model building involves answering the following questions:

(i) What will the state variables be? In the geometric-growth model, there is one state variable: population size. Other models may have multiple state variables, which may refer to multiple species, multiple subpopulations within a species, or other groupings of individuals. There may also be state variables for resources and other abiotic factors.

(ii) What mechanisms will be included in the model? Do we need to include density dependence, stochasticity, species interactions, and so on? For a model with multiple state variables, it can be helpful to construct a flowchart indicating how the different variables influence one another (see, e.g., Chapter 5).

(iii) What will the algebraic structure of the model be? In other words, how do we go from a verbal or graphical statement of (i) and (ii) to a mathematical statement? The kinds of decisions we have to make here include whether to treat variables continuously or discretely, and what precise mathematical forms to choose for functional relationships, such as those that relate population growth to population size. We must also choose names for our variables and parameters. There are no strict rules about such names, but it is sensible to choose names that convey some meaning, e.g., N for number of individuals or t for time.

For each of steps (i)–(iii), we have to make judgements about the complexity of our model. All models make unrealistic assumptions, but how complicated a model should we choose for a given application? Should we strive to include all mechanisms imaginable, thus making our models as complicated and close to reality as possible? Or should we opt for simpler models, since they are at least easier to analyse? There is no straightforward answer to these questions. Theoretical ecologist Simon Levin has mused that "the general philosophy is to incorporate a minimum of necessary detail, complicating the model only when necessary" (Levin 1992). Here "necessary" refers to how much complexity is needed to make the model useful for its intended purpose, whether this purpose is producing insights about a particular ecological phenomenon or generating predictions. Following this advice, we should not throw all conceivable processes into a model at once. Doing so would lead to a model whose behaviour is too complicated to yield general insights, impossible to analyse, and very unlikely to make accurate predictions. In a similar way, a chemist exploring chemical reactions would not throw all reagents into a beaker at once, but would start by exploring the properties of single reagents, then move on to reactions between pairs of reagents, and from there gradually increase the complexity and sophistication of the experiments. By analogy with the prudent chemist, we will usually start with a well-understood model (e.g., the geometric-growth model; Eq. (1.4)) and add in one or two processes at a time to explore their effects on model behaviour.

We emphasise that building models is as much an art as a logical exercise—one that must be taught by way of multiple examples and guiding principles rather than firm rules. Good model-building skills are essential for a theoretical ecologist. No amount of fancy mathematical and computational analysis can save a poorly structured model or one for which the scientific motivation is unclear. In this book, we follow these sentiments and aim not only to impart the technical skills required for model analysis but to help the reader develop the experience and intuition to build models appropriate for the ecological problems at hand. Building mathematical models can seem intimidating at first, but the reader is encouraged to jump in and learn by trial and error (e.g., see Exercises 4 and 5). Even before any analysis takes place, the process of building a mathematical model can itself be instructive in pinpointing aspects of a verbal model that are vague or imprecise. Biologists familiar with field or laboratory work may rightly be apprehensive about jumping into a project without a careful accounting of material costs, health and safety risks, feasibility, and likelihood of success. But the entry barrier to modelling is much lower. Exploring a model that turns out to be flawed may have some minimal costs in terms of time, paper, and ink (or computer power), but even then one is likely to learn something from one's mistakes and can easily screw up the paper and start again.

Let us take the example of the geometric-growth model (Eq. (1.4)) and consider building possible extensions to it. One obvious limitation of this model is its lack of a carrying capacity. Let us then state our research question as, "How does a carrying capacity affect population dynamics?" and be clear that we are looking for general insights rather than predictions for a specific system. For model-building step (i), it would be reasonable to stick with just one state variable, because to explore the effects of a carrying capacity in a minimal way we need only one population. For step (ii), it is clear that we need some kind of density dependence to implement the carrying capacity, but again following the minimalist approach there is no need for other mechanisms. For step (iii), there are multiple possible algebraic choices of the functional form of density dependence, and we will explore some of these in the next chapter.

Exercises

1. The Fibonacci rabbit model is $F_{t+1}=F_t+F_{t-1}$, where F_t is the number of pairs of rabbits at time t (Eq. (1.1)). Solve the Fibonacci rabbit model to find the equilibria (or equilibrium).
2. In the Fibonacci rabbit model, rabbits live forever and after the first month of life each rabbit pair produces one pair of offspring per month. Below are several variations on this model. In each case, write a short verbal description of the biology underlying the model.

(a) $F_{t+1}=F_t+2F_{t-1}$
(b) $F_{t+1}=F_t+F_{t-2}$
(c) $F_{t+1}=\frac{1}{2}F_t+F_{t-1}$
(d) $F_{t+1}=F_t+F_{t-1}+F_{t-2}$
(e) $F_{t+1}=2F_t$

3. Write R code to iterate the models in Exercise 2 and draw time series of the rabbit population size (you may use the code from Box 1.1 as a template).
4. [*] Consider a version of the Fibonacci rabbit model in which 10% of the adult rabbits (i.e., rabbits of reproductive age) die every month. Can you write a mathematical formulation of this model? Is the verbal statement of the model sufficiently precise to allow an unambiguous mathematical formulation?
5. [*] Come up with your own modification of the Fibonacci model. Start with a verbal statement of the model and then write down the mathematical formulation. What biological question would you be answering by analysing your modified model?
6. Consider a model of population growth governed by $N_{t+1}=RN_t$, where $R \geq 0$.

   (a) How does this model relate to the model defined by Eq. (1.4)?
   (b) What is the time-dependent solution of this model? (That is, express N_t explicitly in terms of R and time t.)
   (c) What is the equilibrium of this model?

7. In this chapter we gave an incomplete list of simplifying assumptions underlying the geometric-growth model (Eq. (1.4)). Can you add to this list by identifying more simplifying assumptions, i.e., features of a real population that are ignored by the model?
8. In this chapter, we described a few applications where the geometric-growth model could be useful, despite its simplicity. Can you think of further applications where it might be useful?

References cited

Levin S. A. (1992). The problem of pattern and scale in ecology. *Ecology* 73:1943–1967.

Lubina, J. A., and S. A. Levin (1988). The spread of a reinvading species—range expansion in the California sea otter. *American Naturalist* 131:526–543.

Further reading

Bacaër, N. (2011). *A Short History of Mathematical Population Dynamics*. Springer-Verlag, London.

Index

Pages in *italics* indicate figures and tables; pages in **bold** indicate where a key term is defined or explained.