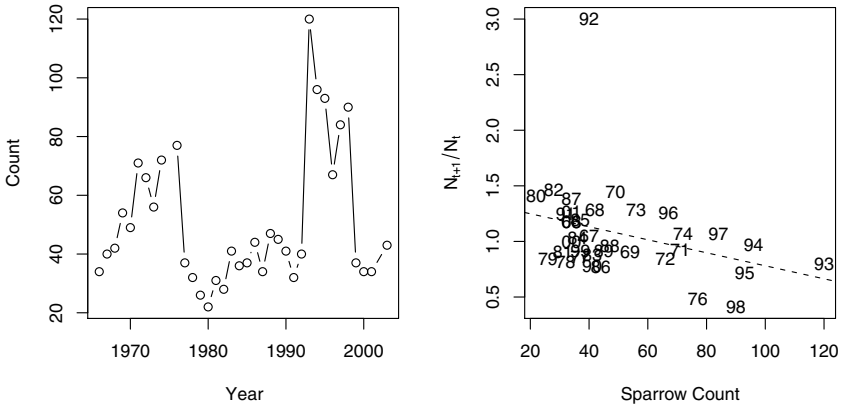
3

Density-dependent Growth

Let's go back to our Song Sparrow (*Melospiza melodia*) data from Chapter 1 on density-independent growth — now we look at *all* the data.



(a) Song Sparrow Counts

(b) N_{t+1}/N_t *vs.* Counts

Fig. 3.1: Song Sparrow *Melospiza melodia* counts from 1966–2003 and the relation between observed counts and annual growth rate determined from those counts, fit with ordinary least squares regression. See Chapter 1 for data source.

In Chapter 1, we modeled Song Sparrow growth without reference to the abundance, or counts, of sparrows, using a small subset of the data. When we look at all the available data (Fig. 3.1a), it seems clear that the annual growth rate ($R_t = N_{t+1}/N_t$) depends on the density of the bird population (Fig. 3.1b). The larger the population, the lower R becomes, until above about 70 counted birds, $R < 1$, indicating population decline. What might limit the population growth of these sparrows? Space available for territories in their successional-scrub type habitat? Availability of seeds, fruits and invertebrates? We don't

M.H.H. Stevens, *A Primer of Ecology with R*, Use R, DOI: 10.1007/978-0-387-89882-7_3,

necessarily know precisely what limits it, but if it is something related to their own abundance, perhaps we can study its *density-dependence*.

Density-dependent population growth is the case where the per capita population growth rate depends statistically on the density of the population. Ecologists consider that negative density-dependence is typically a characteristic of a population undergoing *intraspecific competition*. That is, individuals of the same species are competing for shared resources. So, ... how would we represent this algebraically?[1]

3.1 Discrete Density-dependent Growth

3.1.1 Motivation

To begin, we recall our model of geometric growth for a single time step,

$$N_{t+1} = \lambda N_t. \tag{3.1}$$

We can pull this apart into two sections by recalling that we can decompose λ into two parts, $\lambda = 1 + r_d$, where r_d is the *discrete growth factor*. This allows us to state,

$$N_{t+1} = \lambda N_t = N_t(1 + r_d) = N_t + r_d N_t. \tag{3.2}$$

Here we see the N_{t+1} is equal to the previous year's population, N_t, plus a proportional change, $r_d N_t$. To add density dependence (e.g., Fig. 3.1b), we can build *density-dependence* into that proportional change.

Density dependence means that the population will grow or shrink at a rate that depends on its size. We can imagine that each individual in the population exerts some tiny negative effect on $r_d N_t$, so that the realized per capita growth increment shrinks as N_t grows.

Let us specify that the negative effect of each individual in the population is the same, regardless of how many individuals there are. That is, we could represent this negative effect with a constant, perhaps α, where αN_t is the total negative effect of all individuals in the population. On average, each individual exerts the same negative impact, with a magnitude α, whether there is one individual or 1000 individuals. A particularly convenient way of implementing this is to keep using r_d but to multiply it by a scaling factor. This scaling factor should equal 1 when the population size is zero so that $r_d \times (\text{scaling factor}) = r_d$. The scaling factor should be zero when the population is so big that per capita growth increment is zero, $r_d \times (\text{scaling factor}) = 0$. One such scaling factor looks like this,

$$\text{Per Capita Increment} = r_d(1 - \alpha N_t) \tag{3.3}$$

where α is the per capita negative effect of an individual on the per capita growth increment. As N_t shrinks toward zero, this expression grows toward r_d; as N_t grows, this expression shrinks toward zero.

[1] root: Arabic. *Al-jabr*

At precisely what value of N will per capita growth shrink to zero? We can solve this by noting that r_d is a constant and won't change; all that matters is inside the parentheses. We set the per capita increment equal to zero, and solve for N_t.

$$0 = r_d(1 - \alpha N_t) \tag{3.4}$$

$$0 = r_d - r_d \alpha N_t \tag{3.5}$$

$$N_t = \frac{1}{\alpha}. \tag{3.6}$$

When the population reaches the $N = 1/\alpha$, the growth increment will be zero, and the population will stop growing.

In a sense, the algebraic rearrangement (eq. 3.4) is at the core of theoretical ecology. We began with a set of assumptions (constant r_d and α), assumed a relation between them ($r_d[1 - \alpha N_t]$), and examined one of the consequences: the population will stop changing when it reaches a density of $1/\alpha$, *ceretus paribus*.[2] Welcome to theoretical ecology.

Now instead of density-independent per capita growth, r_d, we have density-dependent per capita growth $r_d(1 - \alpha N_t)$, and our population growth equation becomes,

$$N_{t+1} = N_t + r_d(1 - \alpha N_t)N_t \tag{3.7}$$

This describes *discrete logistic growth.* A common alternative representation uses $1/\alpha$ symbolized as K,

$$N_{t+1} = N_t + r_d\left(1 - \frac{N_t}{K}\right)N_t \tag{3.8}$$

where K represents the carrying capacity. The carrying capacity is the population size at which the per capita growth increment has fallen to zero. The context dictates whether we prefer to represent the scaling factor as the per capita effect, α, or the population carrying capacity, K.

Writing a Function For Discrete Logistic Growth

An R *function* will simplify our explorations. It will return a vector of N, given α, r_d, N_0, and t. The function *arguments* can have defaults (e.g., `t=10`).

```
> dlogistic <- function(alpha = 0.01, rd = 1, N0 = 2, t = 15) {
+     N <- c(N0, numeric(t))
+     for (i in 1:t) N[i + 1] <- {
+         N[i] + rd * N[i] * (1 - alpha * N[i])
+     }
+     return(N)
+ }
```

The function first makes a new vector containing N_0, uses a for-loop to implement eq. 3.7 for each time step, and then returns the vector N.

[2] all else being equal

With discrete logistic growth, if we start with a small population ($N << K$), we will see the population rise and gradually approach K or $1/\alpha$ (Fig. 3.2). We refer to K as an *attractor* because N moves in a deterministic fashion toward K. We explore the meanings of *attractor* and related terms throughout the book.

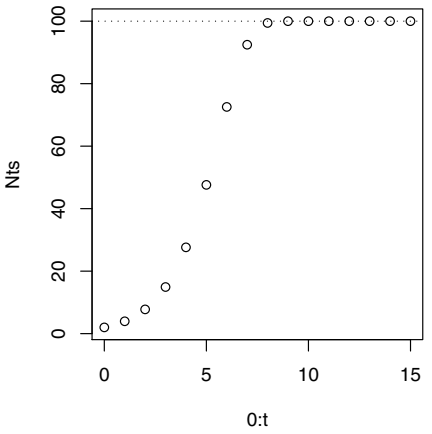


Fig. 3.2: Discrete logistic growth with $r_d = 1$, $\alpha = 0.01$.

Graphing Population Size

We can use the function created above, `dlogistic`, with default settings, to generate a population projection.

```
> Nts <- dlogistic()
```

Now we plot the projection, and put in a dotted line for $1/\alpha$ or K.

```
> t <- 15; a <- 0.01
> plot(0:t, Nts)
> abline(h = 1/a, lty = 3)
```

3.1.2 Relations between growth rates and density

We already know a lot about one description of density dependent growth, the discrete logistic model. In particular, we know that with a constant per capita negative effect, α, the population size at which growth falls to zero is K. Let us explore further how the *per capita* growth increment, and the *population* growth increment, vary with population size.

Casual examination of Fig. 3.2 suggests that the total population growth increment ($\Delta N_t = N_{t+1} - N_t$) starts out small when both t and N are small, accelerates as N grows, and then over time, slows down and reaches an asymptote of

K. Is this changing rate of population growth a function of time, or a function of density? Let us first consider the growth increment as a function of N.

First consider the relation between the population growth increment and population size (Fig. 3.3a). We see it increase as N grows, and then decrease as N approaches K. The pattern is fairly symmetric. That is, it increases and decreases at about the same rates.

Next consider the per capita growth increment ($\Delta N_t/N_t$; Fig. 3.3b). There is a direct linear relation between the per capita growth increment and the size of the population — this is *linear density dependence*. This linear dependence on N comes from our assumption that the per capita negative effect is a constant, α.

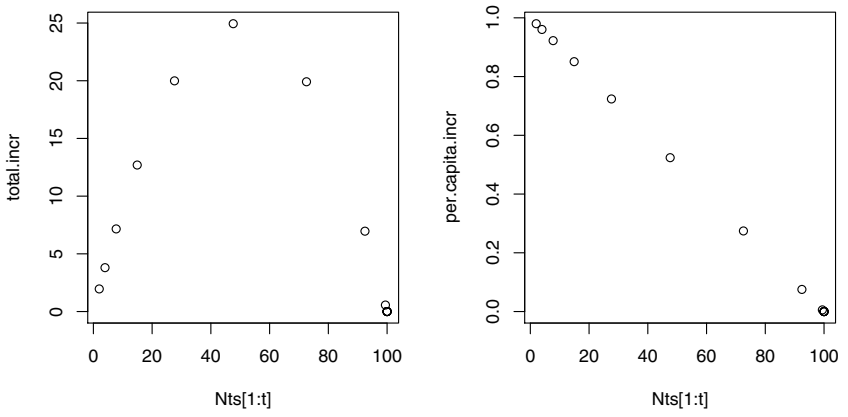
(Per Capita) Population Growth Increment vs. N (Fig. 3.3)

Using the previous projection, we now capture both the total and the per capita growth increment per unit time, from t to $t+1$. We graph these versus N_t, population size at t.

```
> total.incr <- Nts[1:t + 1] - Nts[1:t]
> per.capita.incr <- total.incr/Nts[1:t]

> plot(Nts[1:t], total.incr)

> plot(Nts[1:t], per.capita.incr)
```



(a) Population Growth Increment (b) Per capita Growth Increment

Fig. 3.3: Relations between the total and per capita discrete growth increments and population size.

Let's use a simple analytical approach to understand Figs. 3.3a and 3.3b a little better. Take a look at eq. 3.7. First let's rearrange the expression so that

we can set the *increment* of change equal to zero. How would you do that? Take a minute and try it, before you look at the answer below.

Setting the increment of change equal to zero, we rearrange with the population growth increment on the left, and then set it to zero.

$$N_{t+1} - N_t = r_d N_t (1 - \alpha N_t) \tag{3.9}$$

$$N_{t+1} - N_t = r_d N_t - r_d \alpha N_t^2 \tag{3.10}$$

$$0 = r_d N_t - r_d \alpha N_t^2 \tag{3.11}$$

What do we notice? One observation we could make is that we have a quadratic equation,[3] where the intercept is zero. This tells us that perhaps Fig. 3.3a is symmetric because it is a quadratic expression in terms of N.

What would satisfy this quadratic expression (eq. 3.11), that is, cause the growth increment to equal zero? Well, if r_d or N_t equal zero, those would yield potentially interesting solutions. Assuming neither r_d nor N_t equal zero, we can divide each side by these, and we are left with the solution we found in eq. 3.6, that the growth increment will be zero when $N_t = \frac{1}{\alpha} = K$.

Now let us examine the per capita growth increment (Fig. 3.3b). If we start with the population growth increment eq. 3.9, all we need is to divide through by N_t to get

$$\frac{N_{t+1} - N_t}{N_t} = r_d - r_d \alpha N_t. \tag{3.12}$$

With r_d and α being constants, and N varying, what is this expression? It is the expression for a straight line,[4] just like we observe (Fig. 3.3b). When $N_t = 0$, the per capita increment equals r_d, and when $N_t = 1/\alpha$, the per capita increment is zero. This is precisely where we started when we began the motivation for discrete logistic growth.

3.1.3 Effect of initial population size on growth dynamics

What will be the effect of differences in initial population size? We could approach such a question in at least two ways [21]. For some of us, the simplest way is to play games with numbers, trial and error, plugging in a variety of initial population sizes, and see what happens to the dynamics. If we do this systematically, we might refer to this as a simulation approach. For very complicated models, this may be the *only* approach. Another approach, which is often used in concert with the simulation approach, is the analytical approach. We used this above, when we set the growth equation equal to zero and solved for N. In general, this analytical approach can sometimes give us a definitive qualitative explanation for *why* something happens. This has been used as a justification for using simple models that actually have analytical solutions — they can provide answers [132].

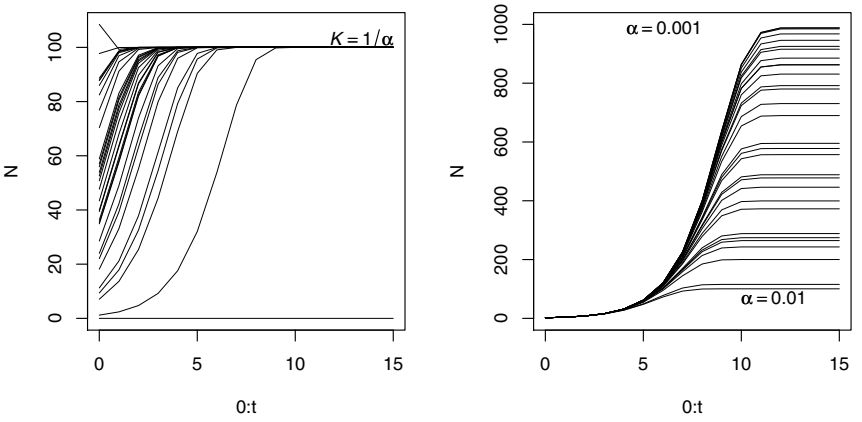
For an analytical approach, first consider the endpoint solutions to the discrete logistc model eq. 3.10. The population will stop changing when $N_t = K$.

[3] $ax^2 + bx + c = 0$

[4] $y = mx + b$.

Note that it does not appear to matter what the initial population size was. The only thing that matters is α. Recall also that the population would not grow if for any reason $N_t = 0$ — the population will be stuck at zero. Based on these analyses, it appears that the only thing that matters is whether the initial population size is zero, or something greater than zero. If the latter, then initial population size appears to have no effect on the eventual population size.

It always pays to check our analytical answer with a brute force numerical approach, so we will use a little simple simulation to see if we are right. In this case, we can vary systematically the initial population size, and see what happens (Fig. 3.4a). What our approach shows us is that regardless of the initial conditions (except zero), N converges on K — K is an attractor. We also might notice that sometimes when $N_0 > K$, it crashes below K before converging on K in the end (more on that later). Last, because there is a qualitative shift in the behavior of the population when $N = 0$, we might want to investigate what happens when N gets very very close to zero. However, in this situation, the analytical solution is so straightforward that it seems convincing that as long as $N > 0$, it will grow toward K.



(a) Variation in N_0 (b) Variation in α

Fig. 3.4: (a) Dynamics due to different initial N (zero was also specifically included, and $\alpha = 0.01$). (b) Dynamics due to different α. All N, except $N = 0$, converge on $K = 1/\alpha$, regardless of the particular value of α ($r_d = 1$).

Numerical Evaluation of Initial Conditions (Fig. 3.4a)

Here we draw randomly 30 N_0 from a uniform distribution between zero and $1.2K$. We also include zero specifically. We then use `sapply` to run `dlogistic` for each N_0, using defaults for the other arguments.

```
> N0s <- c(0, runif(30) * 1.1 * 1/a)
> N <- sapply(N0s, function(n) dlogistic(N0 = n))

> matplot(0:t, N, type = "l", lty = 1, lwd = 0.75, col = 1)
> text(t, 1/a, expression(italic("K") == 1/alpha), adj = c(1,
+     0))
```

A serious simulation might include a much larger number of N_0.

3.1.4 Effects of α

Our conclusions thus far have been based on specific values of α and r_d. Have we been premature? Just to be on the safe side, we should probably vary these also.

What will happen if α varies? This seems easy. First, when N_t is zero, the population growth increment eq. 3.9 is zero, regardless of the magnitude of α. However, when $N_t > 0$, N will increase until it reaches $1/\alpha$ (K; Fig. 3.4b). The outcome seems pretty clear — by decreasing the negative effect of individuals on each other (i.e. decrease α) then the final N increases, and α determines the final N.

Numerical Evaluation of α (Fig. 3.4b)

Here we draw 30 random K from a uniform distribution from 50 to 1000, and convert these to α. We use `sapply` to run `dlogistic` for each α.

```
> a.s <- 1/runif(30, min = 50, max = 1000)
> N <- sapply(a.s, function(a) dlogistic(alpha = a, t = 15))
```

We next plot all populations, and use some fancy code to add some informative text in the right locations.

```
> matplot(0:t, N, type = "l", ylim = c(0, 1000), lty = 1, lwd = 0.75,
+     col = 1)
> text(8, 1/min(a.s), bquote(italic(alpha) == .(round(min(a.s),
+     3))), adj = c(1, 0.5))
> text(10, 1/max(a.s), bquote(italic(alpha) == .(round(max(a.s),
+     3))), adj = c(0, 1.2))
```

Note that we use the minimum and maximum of `a.s` to both position the text, and provide the values of the smallest and largest α.

3.1.5 Effects of r_d

What will variation in r_d do? Probably nothing unexpected, if our exploration of geometric growth is any guide. Our analytical approach indicates that it should have no effect on K (sec. 3.1.2). Nonetheless, let us be thorough and explore the effects of r_d by varying it systematically, and examining the result.

Yikes — what is going on (Fig. 3.5)?. Perhaps it is a good thing we decided to be thorough. These wild dynamics are real — let's go back and look more carefully at r_d.

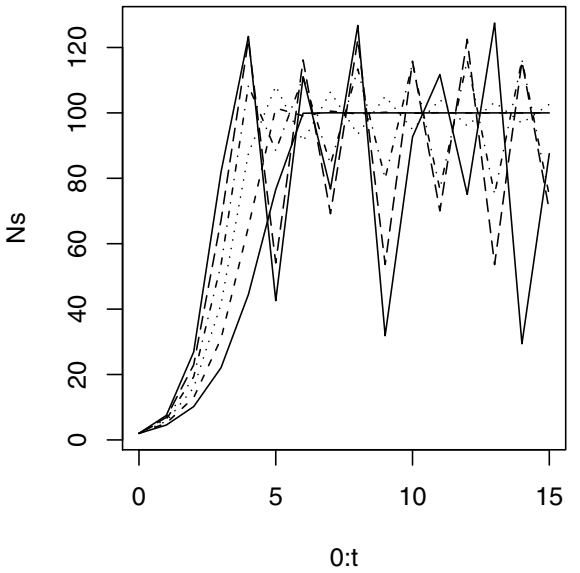


Fig. 3.5: The variety of population dynamics resulting from different values of r_d for the discrete logistic growth model ($r_d = 1, 1.2, \ldots, 3$, $\alpha = 0.01$). See Fig. 3.6 for a more informative view.

Simple Numerical Evaluation of r_d (Fig. 3.5)

Here we vary r_d by creating a short systematic sequence $r_d = 1.3, 1.6, \ldots, 2.8$. We set $t = 50$, and use `dlogistic` to create a trajectory for each of the six r_d.

```
> rd.v <- seq(1.3, 2.8, by = 0.3)
> t <- 15
> Ns <- data.frame(sapply(rd.v, function(r) dlogistic(rd = r,
+     t = t)))

> matplot(0:t, Ns, type = "l", col = 1)
```

Note that many populations do not seem to settle down at K.

If we examine each projection separately, we see a cool pattern is emerging (Fig. 3.6). At the lowest r_d, the population grows gradually toward its carrying capacity, K, and stays there. Once $r_d = 1.6 - 1.9$ it overshoots K just a bit, creating oscillations; these oscillations, however, dampen back down to K. When $r_d = 2.2$, however, the populations seem to bounce back and forth between two values. When $r_d = 2.5$, N bounces around, but now it bounces around between four different values. When $r_d = 2.8$, however, it seems to bounce back and forth around K, but at values that vary every time. This model is just about as simple as a model can be, and includes no random numbers. What is going on?

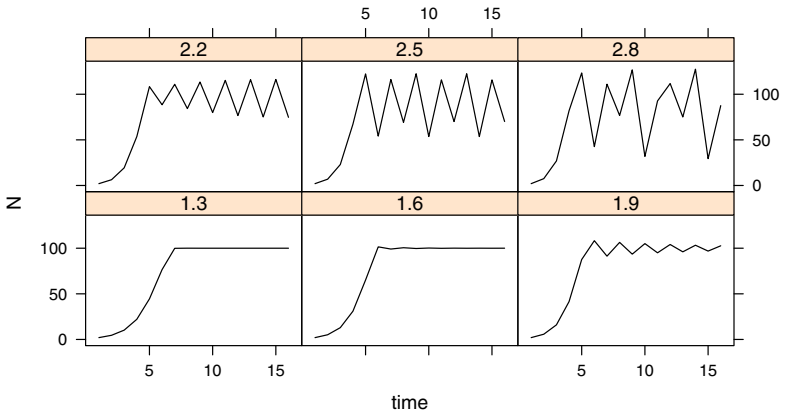


Fig. 3.6: A more informative view of the effects of variation in r_d on population dynamics.

Presentation of Limit Cycles (Fig. 3.6)

First we make a data frame with the six r_d values in the first column, and the respective populations in rows, using `t()` to transpose `Ns`. This puts the data in *wide* format, with a different time step in each column. (This might, for instance, be how you record data in a field experiment with repeated measurements through time).

```
> tmp <- data.frame(rd = as.factor(rd.v), t(Ns))
```

Next, we reshape the data to *long* format, were all N are in the second column, and each is associated with a time step and its r_d value (cols. 3 and 4).

```
> Ns2 <- reshape(tmp, varying = list(2:ncol(tmp)), idvar = "rd",
+     v.names = "N", direction = "long")
> str(Ns2)
```

. . . (output omitted) We plot each trajectory separately using `xyplot` in a different graphics package, `lattice`. Known as a *conditioning* plot, `xyplot` graphs *y vs. x* conditional on *g* (`y ~ x | g`).

```
> library(lattice)
> print(xyplot(N ~ time | rd, data = Ns2, type = "l", layout = c(3,
+     2, 1), col = 1))
```

What is going on is the emergence of *stable limit cycles*, and *chaos*.[5] At low r_d, we have simple asymptotic approach to K. As r_d increases, we see the population overshoot the carrying capacity and exhibit *damped oscillations*. When $2 < r_d < 2.449$, the population is attracted to two-point limit cycles. In this case, *these two points are stable attractors*. Regardless where the population starts out, it is attracted to the same two points, for a given r_d. As r_d increases further, the number of points increases to a four-point limit cycle (e.g., at $r_d = 2.5$), then an eight-point cycle, a 16-point limit cycle, and so on. These points are stable attractors. As r_d increases further , however, stable limit cycles shift into *chaos* ($r_d > 2.57$). Chaos is *a non-repeating, deterministic fluctuating trajectory, that is bounded, and sensitive to initial conditions*.

Robert May [128] shocked the ecological community when he first demonstrated stable limit cycles and chaos using this model. His groundbreaking work, done on a hand calculator, showed how very complicated, seemingly random dynamics emerge as a result of very simple deterministic rules. Among other things, it made population biologists wonder whether prediction was possible at all. In general, however, chaos seems to require very special circumstances, including very high population growth.

Is there a biological interpretation of these fluctuations? Consider some simple environment, in which small vegetation-eating animals with high reproductive rates eat almost all the vegetation in one year. The following year, the vegetation will not have recovered, but the animal population will still be very high. Thus the high growth rate causes a disconnect between the actual population size, and the negative effects of those individuals comprising the population.

[5] Not the evil spy agency featured in the 1960's US television show, *Get Smart*.

The negative effects of the actions of individuals (e.g., resource consumption) are felt by the offspring of those individuals, rather than the individuals themselves. We won't belabor the point here, but it is certainly possible to extend this delayed density dependence to a wide variety of populations. The discrete logistic model has a built in delay, or *time lag*, of one time step, because the growth increment makes a single leap of one time step. This delay is missing from the analogous continuous time model because the growth increment covers an infinity small time step, thanks to the miracles of calculus.[6]

Bifurcations

Up until now, we have examined N as a function of time. We have graphed it for different α and N_0, but time was always on the X-axis. Now we are going to examine N as a function of r_d, so r_d is on the X-axis. Specifically, we will plot the stable limits or attractors *vs.* r_d (Fig. 3.7). What does it mean? For $r_d < 2$, there is only a single N. This is what we mean by a stable point equilibrium, or point attractor — as long as r_d is small, N always converges to a particular point.[7] When $2 < r_d < 2.45$, then all of a sudden there are two different N; that is, there is a two-point stable limit cycle. Note that when $r_d \approx 2$ these oscilliations between the two point attractors around K are small, but as we increase r_d, those two points are farther apart. The point at which the limit cycle emerges, at $r_d = 2$, is called a *bifurcation*; it is a splitting of the single attractor into two attractors. At $r_d \approx 2.45$, there is another bifurcation, and each the two stable attractors split into two, resulting in a total of four unique N. At $r_d \approx 2.53$, there are eight N. All of these points are *periodic* attractors because N is drawn to these particular points at regular intervals. As r_d increases the number of attractors will continue to double, growing geometrically. Eventually, we reach a point when there becomes an infinite number of unique points, *that are determined by* r_d.[8] This completely deterministic, non-repeating pattern in N is a property of *chaos*. Chaos is not a random phenomenon; rather it is the result of deterministic mechanisms generating non-repeating patterns.

[6] A time lag can be explicitly built in to a continuous time model, with only a small effort.

[7] It need not be the same N for each r_d, although in this case it is.

[8] They are also determined by the initial N, but we will get to that later.

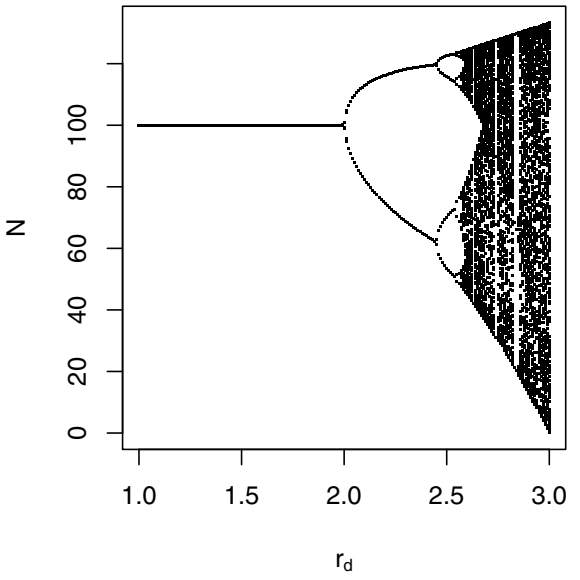


Fig. 3.7: Illustration of the long term dynamics of discrete logistic population growth. When a small change in a continuous parameter results in a change in the number of attractors (e.g. a single point equilibrium to a stable 2-point limit cycle), we call this a bifurcation.

Bifurcation Plot: Attractors as a Function of r_d (Fig. 3.7)

Here we perform more comprehensive simulations, and plot the point and periodic attractors *vs.* r_d. First we pick some constraints for the simulation: the number of different r_d, the sequence of r_d values, and the number of time steps.

```
> num.rd <- 201; t <- 400
> rd.s <- seq(1, 3, length = num.rd)
```

Next we use `sapply` for the simulations (If `num.rd` and `t` are large, this could take a while).

```
> tmp <- sapply(rd.s, function(r) dlogistic(rd = r, N0 = 99,
+     t = t))
```

Next we convert the output to a data frame and stack up the N in one column. We also rename each of the stacked columns, and add new columns for the respective r_d and time steps.

```
> tmp.s <- stack(as.data.frame(tmp))
> names(tmp.s) <- c("N", "Old.Column.ID")
> tmp.s$rd <- rep(rd.s, each = t + 1)
> tmp.s$time <- rep(0:t, num.rd)
```

We save just the later dynamics in order to focus on the N after they have converged on the periodic attractors. Here we select the last 50% of the time steps. (Your figure will look a little different than Fig. 3.7 because I used more r_d and time steps.)

```
> N.bif <- subset(tmp.s, time > 0.5 * t)
> plot(N ~ rd, data = N.bif, pch = ".", xlab = quote("r"["d"]))
```

There has been a great deal of effort expended trying to determine whether a particular model or real population exhibits true chaos. In any practical sense, it may be somewhat unimportant whether a population exhibits true chaos, or merely a higher order periodic attractor [49]. The key point here is that very simple models, and therefore potentially simple mechanisms, can generate very complex dynamics.

Sensitivity to initial conditions

Another very important characteristic feature of chaotic populations is that they are very sensitive to initial conditions. Thus emerges the idea that whether a butterfly in Sierra Leone flaps its wings twice or thrice may determine whether a hurricane hits the southeastern United States in New Orleans, Louisiana, or in Galveston, Texas.[9]

If we generate simulations where we vary initial population size by a single individual, we find that this can have an enormous impact on the similarity of two populations' dynamics, and on our ability to predict future population sizes (Fig. 3.8). Note how the populations start with similar trajectories, but soon diverge so that they experience different sequences of minima and maxima (Fig. 3.8). This is part of what was so upsetting to ecologists about May's 1974 paper — perhaps even the simplest deterministic model could create dynamics so complex that we could not distinguish them from random [128]. Over time, however, we came to learn that (i) we could distinguish random dynamics from some chaos-like dynamics, and (ii) the hunt for chaos could be very exciting, if most frequently disappointing [8, 39, 90].

Sensitivity to Intitial Conditions

We start with three populations, all very close in initial abundance. We then propogate with a r_d to generate chaos for 100 time steps.

```
> N.init <- c(97, 98, 99); t <- 30
> Ns <- sapply(N.init, function(n0) dlogistic(rd = 2.7, N0 = n0,
+     t = t))
```

Now we would like to graph them over the first 12 times, and look at the correlations between N_1 and the other two populations.

```
> matplot(0:t, Ns, type = "l", col = 1)
```

Boundedness, and other descriptors

One last issue that we should note is the extent to which our populations are *bounded*. A population may have complex dynamics, but we may be able to characterize a given population by its upper and lower bounds. In spite of the

[9] Clearly, this suggests that the solution to increased storm severity due to global warming is to kill all butterflies.

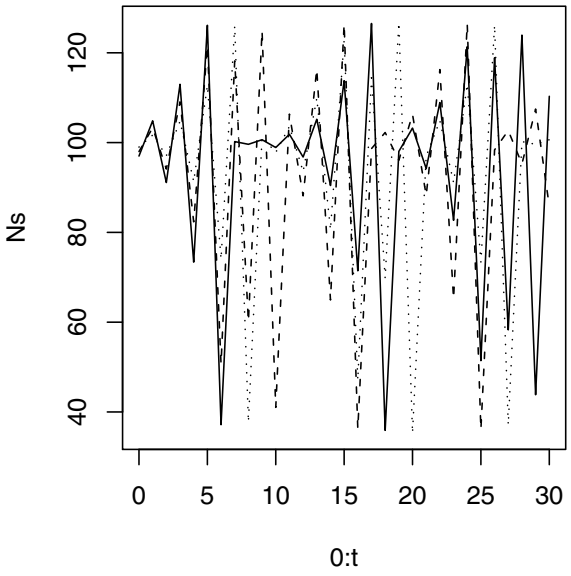


Fig. 3.8: Effects of differences in initial population size on the short term and long term dynamics, and their correspondence, of three populations.

differences created by the initial conditions, the upper and lower bounds of our chaotic populations were very similar (Fig. 3.8). Note also as r_d increases (Fig. 3.7) the oscillations increase very systematically.

In general, we can describe many characteristics of populations, even if we cannot predict exactly what the population size will be five years hence. For instance, we can describe the shape of density dependence (linear, nonlinear), and characteristics of N, such as the average, the variance, the periodicity, the upper and lower bounds, and the color (i.e. degree of temporal auto-correlation). Indeed, these characteristics may vary among types of organisms, body sizes, or environments.

3.2 Continuous Density Dependent Growth

The classic version of continuous density dependent growth [93] is the continuous logistic growth equation, the continuous version of eq. 3.8 above,

$$\frac{\mathrm{d}N}{\mathrm{d}t} = rN\left(\frac{K-N}{K}\right) \tag{3.13}$$

which, as we saw above, is no different than

$$\frac{\mathrm{d}N}{\mathrm{d}t} = rN\left(1-\alpha N\right). \tag{3.14}$$

Take a moment to see the parallels between these and eq. 3.7.

3.2.1 Generalizing and resimplifying the logistic model

By now you might be wondering why we are studying such simplistic models that make such unrealistic assumptions. Let's use the next couple of pages to figure out how we might relax some of these assumptions, and generalize eq. 3.14.

First let's realize that the density independent per capita growth rate, r, is really a net rate — it is the difference between the density independent, instantaneous birth and death rates, b and d. If we start with per capita logistic growth, the generalization would look like this.

$$\frac{\mathrm{d}N}{N\mathrm{d}t} = r(1-\alpha N) \tag{3.15}$$

$$= (b-d)(1-\alpha N) \tag{3.16}$$

$$= b - d - b\alpha N + d\alpha N \tag{3.17}$$

where $b - d = r$.

Note that the positive, density-independent effect on growth rate of b is counterbalanced a bit by a negative density-dependent effect of b. As N increases, increasing births tend to rein in growth rate a bit, because more births mean a larger negative density-dependent effect. Similarly, mortality, d includes a small *positive* density-dependent effect that helps enhance growth rate because it reduces the negative effect of αN — death frees up resources.

Density dependence is the effect of density on growth rate, and thus far we have let that be $1 - \alpha N$. Let's represent this as the function $F(N) = 1 - \alpha N$. We can now note that α is also a net effect — N will have separate effects on the birth and death rates, and α could be just the sum of more arbitrary constants (e.g., $\alpha N = (x + y)N$); $1 - \alpha N$ is merely the simplest form of $F(N)$. We can generalize further and let density affect the birth and death rates separately,

$$\frac{\mathrm{d}N}{N\mathrm{d}t} = rNF(N) \tag{3.18}$$

$$F(N) = B(N) + D(N) \tag{3.19}$$

We might anticipate that density has no effect on death rates when N is low, because there is no particular limiting factor (Fig. 3.9a). Paradoxically, however, when N becomes large, the ensuing mortality benefits growth rate a bit because death frees up limiting resources (Fig. 3.9a). Such an idea might have the simple form

$$D(N) = gN^2 \tag{3.20}$$

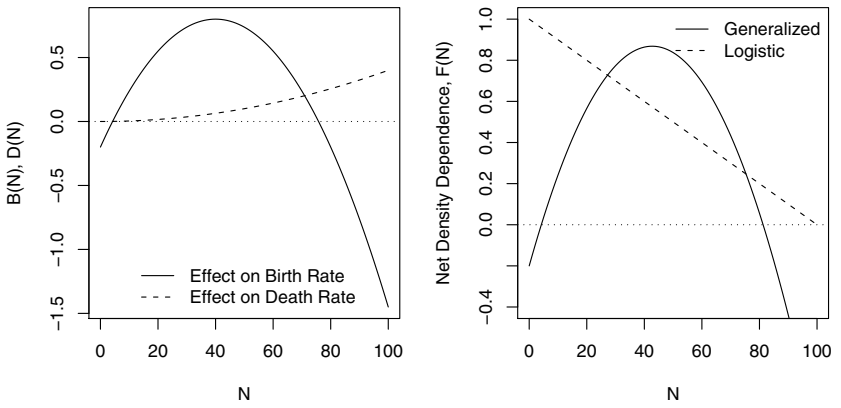
where g is a constant. This function starts out at zero (no effect) and increases rapidly (Fig. 3.9a). This means that at low density, mortality does not influence growth rate, whereas at high density, mortality enhances growth rate.

The density dependence for birth rates could also be more complicated. The *Allee effect* [192] arises when a population gets very small and mating success declines because mates can't find each other, leading to a large negative per capita impact of N. We can describe this with a quadratic function,

$$B(N) = -aN^2 + eN - f \tag{3.21}$$

where a, e, and f are constants. Here, the negative effect of N on birth rates, $B(N)$, will be important at low N because when $N = 0$, $B(N) = -f$. The negative effect will also be large at very high N because $-aN^2$ becomes a large negative number at large N (Fig. 3.9a). The point here is that we can alter density dependence to incorporate specific biology.

If we sum the density dependences, we have a function $F(N) = B(N) + D(N)$ that is analogous to $1 - \alpha N$, but which is a different shape, because it describes a particular underlying biology (Fig. 3.9b). Note the differences between the simple linear density dependence of the logistic model, and the nonlinear density dependence of our generalization.



(a) Birth and Death Rates (b) Net Density Dependence

Fig. 3.9: Elaborating on simple logistic growth. (a), a specific example of a generalized density dependence via effects on birth rate and death rate. (b) Net density dependence for our generalized model and logisitc linear density dependence.

Density Dependence on Birth and Death Rates (Figs. 3.9a, 3.9b)

To make expressions for eqs. 4.24, 4.17, use `expression`.

```
> B.N <- expression(-a * N^2 + e * N - f)
> D.N <- expression(g * N^2)
```

We then provides constants and evaluate the expressions to plot the density dependence on birth and death rates.

```
> a <- 1/1600; e <- 80 * a; f <- 0.2; g <- 4e-05; N <- 0:100

> plot(N, eval(B.N), type = "l", ylab = "B(N), D(N)")
> lines(N, eval(D.N), lty = 2)
> abline(h = 0, lty = 3)
> legend("bottom", c("Effect on Birth Rate", "Effect on Death Rate"),
+     lty = 1:2, bty = "n")
```

The sum of these two rates is merely density dependence,

```
> plot(N, eval(B.N) + eval(D.N), type = "l", ylim = c(-0.4,
+     1), ylab = "Net Density Dependence, F(N)")
> abline(h = 0, lty = 3)
```

and we can compare this to linear density dependence of the logistic model $(1-\alpha N)$. If $\alpha = 0.01$, we have

```
> curve(1 - 0.01 * x, 0, 100, lty = 2, add = T)
> legend("topright", c("Generalized", "Logistic"), lty = 1:2,
+     bty = "n")
```

The extreme flexibility of eq. 3.15 has a serious downside—we now have to estimate more parameters. Clearly, if $r = 0.5$, then b and d can take on any values, provided their difference is 0.5. We can make the same argument with regard to linear density dependence as well — $B(N)$ and $D(N)$ could take an infinite variety of forms, provided their sum equals $1 - \alpha N$.

In an attempt to simplify our lives a bit, and in the absence of any additional information, let us impose the simplest function for B and D, a linear constant

$$\frac{\mathrm{d}N}{N\mathrm{d}t} = (b - m_b N) - (d - m_d N) \tag{3.22}$$

where m_b and m_d are constants that describe the effect of an additional individual on the base density-independent per capita birth and death rates, respectively. If we really wanted to simplify our lives, we could let $m_b + m_d = m$, $b - d = r$, and simplify to

$$\frac{\mathrm{d}N}{N\mathrm{d}t} = r - mN \tag{3.23}$$

Thus, the *net* effect of N is linear, even though the effects of N on birth and death rates may not be. For reasons that probably have more to do with history than with mathematics, we create another constant, $\alpha = m/r$, and we find ourselves back to

$$\frac{\mathrm{d}N}{N\mathrm{d}t} = r(1 - \alpha N)\,. \tag{3.24}$$

Let's review the circular path of elaboration and simplification upon which we have travelled. We started with a historical representation of logistic growth, using K to represent an equilibrium population size toward which N is attracted. We represented this in a tad more mechanistic fashion (I would argue) by rephrasing the limitation factor in terms of a per capita effect of an individual on the growth rate. We then expanded and generalized this expression to see that the *net* effects, r and α, could each be thought of as two components related to birth rates and death rates. We further expanded on density dependence, allowing it to be curvilinear. We then collapsed this back to a simple difference ($r - mN$, eq. 3.23) and finally recast it in the original expression.

The integral of logistic growth

We might as well add that there is an expression for N_t, analogous to $N_t = N_0 \exp(rt)$ for exponential growth. If we integrate eq. 3.24, we have

$$N_t = \frac{N_0 e^{rt}}{1 + \alpha N_0 (e^{rt} - 1)}. \tag{3.25}$$

Note the resemblance to the exponential equation. We have the exponential equation in the numerator, divided by a correction factor which starts at 1 when $t = 0$. The correction factor then grows with time, at a rate which is virtually the same as exponential growth but adjusted by α and scaled by N_0. Thus when $t = 0$, and if N_0 is small, we have near-exponential growth, decreased merely by a factor of αN_0. We will take advantage of this integral below, when we describe some data in terms of the logistic growth model.

3.2.2 Equilibria of the continuous logistic growth model

As we did with the discrete model, we can determine the *equilibria* of the continuous logistic growth model, that is the values of N for which $\mathrm{d}N/\mathrm{d}t = 0$. Consider again eq. 3.24

$$\frac{\mathrm{d}N}{\mathrm{d}t} = rN(1 - \alpha N). \tag{3.26}$$

What values of N will cause the growth rate to go to zero, thus causing the population to stop changing? As with the discrete model, we would find that if $N = 0, 1/\alpha$, the growth rate would be zero, and the population stops changing. Therefore the equilibria for continuous logistic growth are $N^* = 0, 1/\alpha$.

3.2.3 Dynamics around the equilibria — stability

The *stability* of a system[10] is a measure of how much it tends to stay the same, in spite of external disturbances or changes in the state of the system.

[10] A system might be nearly any set of interacting parts, such as a population, or an economy, or the internet.

The term *stability* has been given many more specific meanings, including resilience, resistant, reactivity, and permanence. We won't go into these here, but one could consider them different facets of stability [25].[11] We will focus on resilience, the tendency for a population to return an equilibrium, or be *attracted toward* an equilibrium, if it is perturbed from it.

Consider the stability of a marble inside a wok. If the wok doesn't move, then the marble just sits in the lowest spot on the bottom. If the wok is bumped, the marble jiggles and rolls around a bit, but settles back down to the bottom. This is a stable system because there is a point at the bottom of the bowl (the attractor) toward which the marble rolls — all points inside the wok allow the marble to find the lowest point (the attractor). For this reason, we call the collection of points on the inside surface of the wok the *basin of attraction*. The steeper the sides, the more quickly the marble moves toward the bottom. This rate is referred to as its *resilience*.

To translate the notion of a basin into a population, let the position of marble inside the wok be the value of N at any particular point. Bumping the wok is any disturbance that causes a change in N. The slope of the sides of the wok reflects the biology and its mathematics that cause N to move quickly or slowly back toward the bottom. For the logistic model, this rate is determined by r and α, and also by N itself. The attractor at the very bottom of the bowl is the population's carrying capacity, $K = 1/\alpha$.[12]

When we imagine a marble in a wok, it becomes easy to envision K as an attractor at the bottom of the bowl. That is why we consider the carrying capacity a stable equilibrium point, or attractor, even if other factors, such as a disturbance, may cause N to deviate from it.

The equilibrium we have focused on has been K, but recall that $N = 0$ is also an equilibrium. This equilibrium is actually the edge of the wok — the slightest jiggle, and the marble falls in and moves toward the center of the wok, K. The biological analog of this "jiggle" is the additional of one or a very small numbers of individuals to an otherwise extinct population. For example, consider that a sterile petri dish with nutrient agar has an *E. coli* population size of zero. If that *E. coli* $N = 0$ gets "perturbed" with a single added cell, the population will move quickly toward its carrying capacity K. In this sense, $N = 0$ is an equilibrium, but it is an *unstable* equilibrium. We also call such an unstable equilibrium a *repeller*.

Analytical linear stability analysis

We are interested in the dynamics or stability of N at each of the equilibria, N^*. Here we use *analytical linear stability analysis* to show mathematically what we

[11] Resistance refers to the tendency to resist change in the first place; resilience refers to the long term rate of return to an equilibrium; reactivity refers to the rate of change immediately following such a disturbance [148], and permanence is the degree to which a system does not simplify by entering one of its boundary equilibria.

[12] The marble's inertia, which contributes to its momentum, is analogous to a time lag in density dependence that we saw in the discretet model; with a time lag, a population can overshoot the equilibrium.

described above with the marble and the wok. While somewhat limited in its scope, linear stability is nonetheless a powerful technique for dynamic model analysis.

In a nutshell, what we do is determine whether the growth rate, which is zero at each equilibrium, becomes positive or negative in response to a small change in N. That is, if N takes a tiny step away from the equilbrium, does that tiny step shrink, or grow? If a tiny step results in a shrinking of that step, back toward the equilibrium, that demonstrates stability. If a tiny step results in growth and a bigger step, that demonstrates instability. That is what analytical stability analysis tells us.

Consider a plot of the growth rate, $\mathrm{d}N/\mathrm{d}t$ *vs.* N (Fig. 3.10). The equilibria, N^*, are the N (x-axis) at which $\mathrm{d}N/\mathrm{d}t = 0$ (points a, d). Note where population growth rate is positive and where it is negative. What will happen to this population if it finds itself at $N = 50$? It will grow, moving along the x-axis, until population growth rate slows so much that it comes to rest where $N = 1/\alpha = K = 100$. Thus, N changes until it converges on the equilibrium, N^*, where $\mathrm{d}N/\mathrm{d}t = 0$. Alternatively at $N = 110$, $\mathrm{d}N/\mathrm{d}t$ is negative, and so N shrinks back down to K (point d). This return toward K means that K is an attractor and a stable equilibrium.

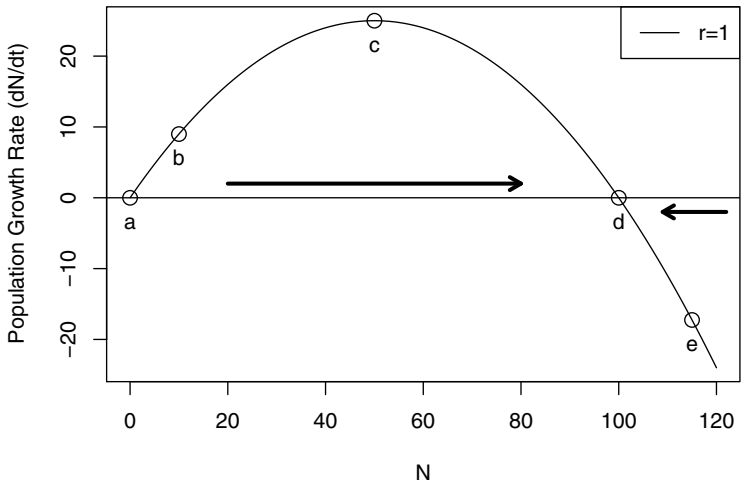


Fig. 3.10: Population growth rate, $\mathrm{d}N/\mathrm{d}t$, as a function of N. Points a–e are labelled for convenience. At values of N associated with points a and d, growth rate (here labelled as $F(N)$) equals zero. At values of N associated with b and c growth rate is positive, and for e growth rate is negative. *Note this is growth rate as a function of* N (time is not explicitly part of this graph).

Next, consider $N = 0$, at point a; it cannot change on its own. However, if "perturbed" at all, with the addition of one or more individuals, then this

"perturbation" will grow, sending N across the x-axis, away from $N = 0$, toward $N = K$. This stasis at $N = 0$, and movement away from $N = 0$ with the slightest perturbation means that $N = 0$ is a repeller and an unstable equilibrium.

Linear stability analysis will calcculate the degree of attraction or repulsion.

Growth rate vs. N

We first define an expression, and constants.

```
> pop.growth.rate <- expression(r * N * (1 - alpha * N))
> r <- 1; alpha <- 0.01; N <- 0:120
```

A basic plot.

```
> plot(N, eval(pop.growth.rate), type = "l",
+      ylab = "Population Growth Rate (dN/dt)", xlab = "N")
> abline(h = 0); legend("topright", "r=1", lty = 1)
```

Add a few points with labels,

```
> N <- c(0, 10, 50, 100, 115)
> points(N, eval(pop.growth.rate), cex = 1.5)
> text(N, eval(pop.growth.rate), letters[1:5], adj = c(0.5, 2))
```

and arrows for the direction of change in N.

```
> arrows(20, 2, 80, 2, length = 0.1, lwd = 3)
> arrows(122, -2, 109, -2, length = 0.1, lwd = 3)
```

How fast will N return to $N^* = K$, if perturbed? Let's start by imagining that we have two populations with different intrinsic rates of increase ($r = 1, 0.5$), and focus in on the growth rate at $N^* = K$ (Fig. 3.11a). For which one of these populations will $\mathrm{d}N/\mathrm{d}t$ go to zero more quickly? If each population loses a few individuals and declines by x to $N^* - x$, which population will return to N^* more quickly? Not surprisingly, it is the population with the higher r — when $r = 1$, its population growth rate (y-axis) is greater than when $r = 0.5$, and therefore will return at a faster rate.

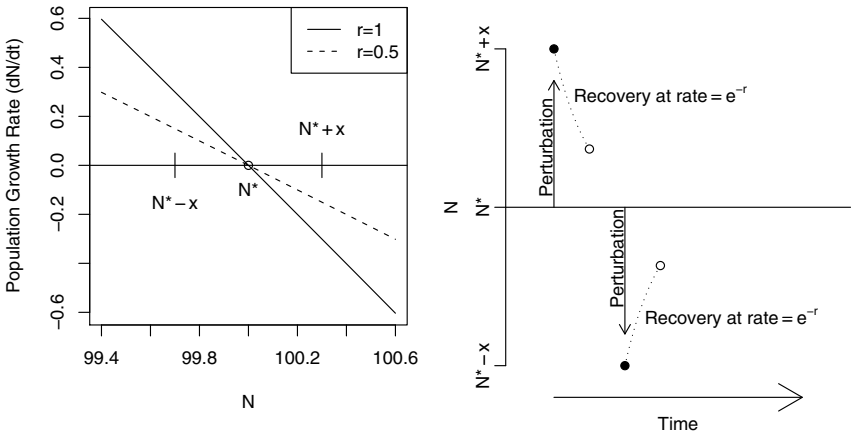
Note also that this same population ($r = 1$) has a negative growth rate of greater magnitude at $N^* + x$. Regardless of whether it gains or loses individuals, it will return to N^* more quickly than the population with the lower r.

How do we quantify the rate of return at N^* so that we can compare two or more populations? The slope of the curve at N^* quantifies the rate of return, because the slope is the exponential return rate. To calculate the slope of a curve we use calculus, because the slope of a curve is a derivative. In this case, we need the derivative of the growth rate ($\mathrm{d}N/\mathrm{d}t$) with respect to the variable on the x-axis, N. Such a derivative is called a *partial derivative*. Therefore we need the partial derivative of growth rate, with respect to N. This will be the slope of the growth rate with respect to N.

To find the partial derivative, let us denote the growth rate as $\dot{N}$ ("N-dot"),

$$\dot{N} = rN(1 - \alpha N). \tag{3.27}$$

$\dot{N}$ is a common symbol for a time derivative such as a growth rate.



(a) Growth Rate *vs.* N (b) Exponential Recovery

Fig. 3.11: Very close to an equilibrium, a population recovers from a perturbation, moving toward the equilibrium attractor, at rate e^{-r}. (a), Slopes of population growth *vs.* N near the equilibrium; around the equilibrium attractor, small decreases in N lead to positive growth rate, whereas small increases lead to negative growth rate; the population with the steeper slope changes faster. (b), Regardless of the particular r of a population, a population recovers from a perturbation, moving toward the equilibrium attractor, at rate e^{-r}.

Given $\dot{N}$, its partial derivative with respect to N is

$$\frac{\partial \dot{N}}{\partial N} = r - 2r\alpha N. \tag{3.28}$$

Further, we are interested in the equilibrium, where $N = 1/\alpha$. If we substitute this into eq. 3.28 and simplify, this reduces to

$$\frac{\partial \dot{N}}{\partial N} = -r\,. \tag{3.29}$$

At $N^* = 1/\alpha$, the slope is $-r$. *This negative slope at the equilibrium demonstrates the stability of this equilibrium.*

Near N^*, population growth rate is $-rx$.[13] That is, the rate of change of x (Figs. 3.11a, 3.11b) is

$$\frac{\mathrm{d}x}{\mathrm{d}t} = -rx \tag{3.30}$$

This allows us to pretend that the perturbation will diminish exponentially, because the growth rate is constant. We say "pretend," because we realize that this rate applies only in the small region where we can assume the slope is a straight line. We can describe the size of x at any time t by integrating this well-known differential equation with respect to time as

[13] Because $\Delta y/\Delta x = -r$, i.e. the change in y equals the change in x times the slope.

$$x_t = x_0 e^{-rt} \tag{3.31}$$

where x_0 is the size of the initial displacement relative to the equilibrium, and x_t is the size of the displacement at time t relative to the equilibrium.[14]

If we choose an x_t carefully, and do so for any and all such analyses, we can determine the time required to reach x_t and we call that time the *characteristic return time*. To determine the characteristic return time, we will let $x_t = x_0/e$, thus defining the characteristic return time as the amount of time required to reduce the perturbation by 63% (i.e. $1/e$). We can solve for t by dividing through by x_0 and taking the log of both sides,

$$\frac{x_0}{e} = x_0 e^{-rt} \tag{3.32}$$

$$\log\left(e^{-1}\right) = -rt \tag{3.33}$$

$$t = -\frac{1}{(-r)}. \tag{3.34}$$

Thus we see return time, t, here is a positive number, with the same units as r, and depends on the slope of the the growth rate with respect to N. It also depends on the assumption of linearity very close to the equilibrium.

To review, we took the partial derivative of $\mathrm{d}N/\mathrm{d}t$ with respect to N, and then evaluated the partial derivative at N^* to get the rate of change of the perturbation, x. A negative value indicates that the perturbation will decline through time, indicating a stable equilibrium or attractor. A positive value would indicate that the perturbation will grow over time, thus indicating an unstable equilibrium, or a repellor.

Symbolic differentiation

We can use R's minimal symbolic capabilities to get derivatives. Here we get the partial derivative and evaluate for the two equilibria (Fig. 3.10).

```
> dF.dN <- deriv(pop.growth.rate, "N")
> N <- c(0, 1/alpha)
> eval(dF.dN)

[1] 0 0
attr(,"gradient")
      N
[1,]  1
[2,] -1
```

The first value, 1, corresponds to the first value of N, which is 0. Because it is positive, this indicates that the perturbation will increase with time, meaning that $N = 0$ is a repellor. The second value, −1, is negative, and so indicates that the perturbation will decrease with time, meaning that $N = 1/\alpha$ is an attractor.

[14] We also did this in Chapter 1, for exponential growth.

3.2.4 Dynamics

How does the continuous logistic growth equation behave? It is very well behaved, compared to the discrete version. It has a very characteristic shape (Fig. 3.12a), and regardless of r or N_0, it moves converges boringly toward $K = 1/\alpha$ (Fig. 3.12b).

Function for an ODE

Making a function to use with R's ODE solver is pretty easy, provided we follow the rules (see Appendix, secs. , B.10). To make the code transparent, I translate the vector parameters and the vector of populations (in this single species model, we have only one population).

```
> clogistic <- function(times, y, parms) {
+     n <- y[1]
+     r <- parms[1]
+     alpha <- parms[2]
+     dN.dt <- r * n * (1 - alpha * n)
+     return(list(c(dN.dt)))
+ }
```

We create vectors for the parameters and the initial densities for all of the populations in the model. We also specify the time.

```
> prms <- c(r = 1, alpha = 0.01); init.N <- c(1); t.s <- seq(0.1,10, by = 0.1)
```

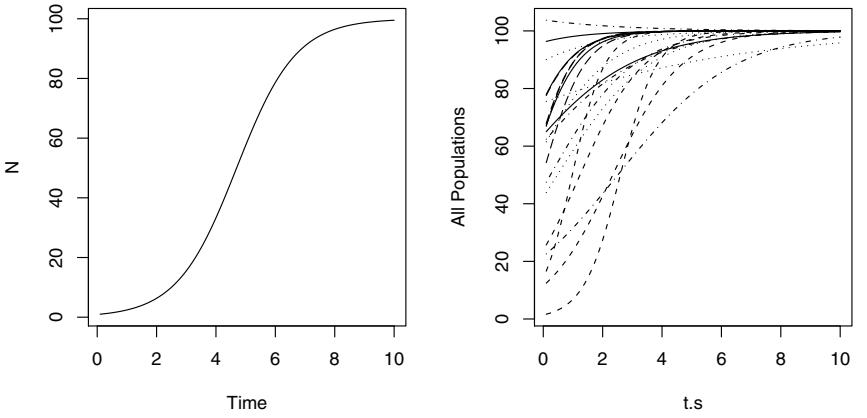
We load the `deSolve` library, and run `ode`. The output is a matrix that includes the time steps and the N (Fig. 3.12a).

```
> library(deSolve)
> out <- ode(y = init.N, times = t.s, clogistic, parms = prms)

> plot(out[, 1], out[, 2], type = "l", xlab = "Time", ylab = "N")
```

So why is the continuous version so boring, while the discrete version is so complex? Remind yourself what is going on with the discrete version. The model could only take steps from one generation to the next. The step into generation $t + 1$ was a function of N at t, and not a function of N_{t+1}. Therefore the rate of change in N was not influenced by a contemporaneous value of N. There was a delay between N and the effect of N on the population growth rate. For many organisms, this makes sense because they undergo discrete reproductive events, reproducing seasonally, for instance. In contrast, the continuous logistic population growth is always influenced by a value of N that is updated continuously, instantaneously. That is the nature of simple differential equations. They are instantaneous functions of continuous variables. We can, if we so choose, build in a delay in the density dependence of continuous logistic growth. This is referred to as "delayed density dependence" or "time-lagged logistic growth"

$$\frac{\mathrm{d}N}{\mathrm{d}t} = rN(1 - \alpha N_{t-\tau}) \tag{3.35}$$



(a) One Population (b) Many Different Populations

Fig. 3.12: Dynamics of continuous logistic population growth. (a) The characteristic shape of logistic growth. (b) Regardless of N_0 or r, populations converge slowly (small r) or quickly (high r) on $K = 1/\alpha$.

where τ is the degree of the delay, in time units associated with the particular model. Just as with the discrete model, the dynamics of this continuous model can get very complicated, with sufficient lag and sufficient r.

Plotting Random Populations ((Fig. 3.12b))

We use the above function to create 20 populations with different traits. We start with an empty matrix, and then for each of the populations, we draw random N_0 and r, run the ODE solver, keeping just the column for N. Last we plot the output.

```
> outmat <- matrix(NA, nrow = length(t.s), ncol = 20)
> for (j in 1:20) outmat[, j] <- {
+     y <- runif(n = 1, min = 0, max = 120)
+     prms <- c(r = runif(1, 0.01, 2), alpha = 0.01)
+     ode(y, times = t.s, clogistic, prms)[, 2]
+ }

> matplot(t.s, outmat, type = "l", col = 1, ylab = "All Populations")
```

3.3 Other Forms of Density-dependence

There are commonly used types of density-dependence other than logistic. Some, like the Richards model, are more flexible and have more parameters. Others, like the Gompertz model, are used more widely in other fields, such as cancer research for tumor growth, von Bertlanaffy for body size growth, and the Ricker model for fisheries; the Richards model provides even more flexibility, at the

cost of more paramteters. Here we explore a simple extension of the logistic model, the *theta-logistic* model, which adds a parameter to increase flexibility and generality.

$$\frac{\mathrm{d}N}{\mathrm{d}t} = rN\log\left(1-(\alpha N)^{\theta}\right) \tag{3.36}$$

Here θ is strictly positive ($\theta > 0$); $\theta = 0$ means zero growth, and $\theta < 0$ would mean negative growth below K, and unbounded positive growth above K. Approximations of eq. 3.36 that allow $\theta \leq 0$ are possible [188], but the interpretation becomes strained.

Theta-logistic function

Here we make a function that we can use with `ode`, the numerical integration function.

```
> thetalogistic <- function(times, y, parms) {
+     n <- y[1]
+     with(as.list(parms), {
+         dN.dt <- r * n * (1 - (alpha * n)^theta)
+         return(list(c(dN.dt)))
+     })
+ }
```

Using `with()` and `as.list()` creates an environment in which R will look inside `parms` for named elements. This will work as long as we name the parameters in `parms`.

By varying θ, we can change the linear density dependence of the simple logistic model to curvilinear density dependence (Fig. 3.13a). This curvilinearity arises because when $\alpha N < 1.0$ (i.e. $0 < N < K$),

$$(\alpha N)^{\theta} < \alpha N, \quad \theta > 1 \tag{3.37}$$

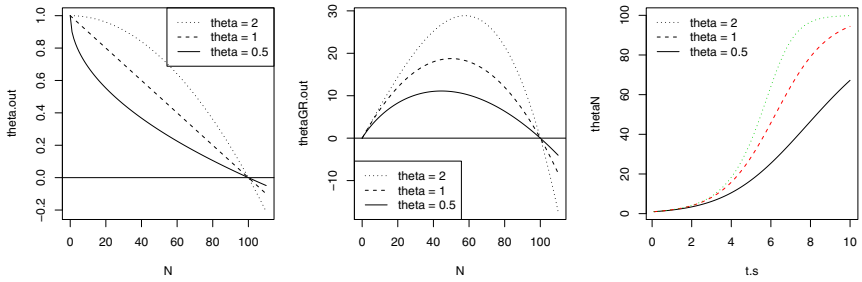
$$(\alpha N)^{\theta} > \alpha N, \quad \theta < 1. \tag{3.38}$$

In contrast, when $\alpha N = 1.0$, then $(\alpha N)^{\theta} < \alpha N$. This means that θ does not affecct K.

When $\theta > 1$, this weakens density dependence at low N, so the population grows faster than logistic, all esle being equal. When $\theta < 1$, this strengthens density dependence at low N, causing the population to grow more slowly than logistic, all else being equal.

The effects of θ on density dependence controls the shape of relation between growth rate *vs.* N (a.k.a. the production function, Fig. 3.13b). First, note that for a given r, growth rate for $N < K$ increases with θ. Second, note that the position of peak of the production function shifts to the right as θ increases. That is, as θ increases, the N at which the maximum growth rate occurs also increases. If we wanted to shift the peak growth rate to a higher N without also increasing the height of the peak, we could decrease r simultaneously.

We could speculate on biological meaning of θ and the shape of the denisty dependence. For instance, very high θ suggests a threshold, wherein the population exhibits little density dependence until very close to K. Perhaps this is



(a) Density Dependence (b) Growth Rate (c) N Dynamics

Fig. 3.13: Theta-logistic growth. In a review of population dynamics, Sibly et al. [188] use theta-logistic density dependence, $1-(N/K)^{\theta}$, to show that populations most frequently have a concave-up $\theta < 1$ pattern.

related to territoriality, or a spatial refuge from predators. Alternatively, low θ might suggest resource preemption, wherein a small number of individuals can sequester large amounts of resources, but increasing N results in reduced preemption. For organisms with very plastic body sizes (plants, fish), this could mean that at low N, average body size is large, but as $N \to K$, average body size decreases. While θ, *per se*, is devoid of mechanism, discovering the magnitude of θ for different types of species [188] could lead to the generation of new testable hypotheses about what controls populations.

Theta-logistic density dependence

We first graph theta-logistic, for $\theta < 1$, $\theta = 1$, and $\theta > 1$ (Fig. 3.13a).

```
> r <- 0.75; alpha <- 0.01; theta <- c(0.5, 1, 2);
> N <- 0:110
> theta.out <- sapply(theta, function(th) {
+     1 - (alpha * N)^th
+ })
> matplot(N, theta.out, type = "l", col = 1)
> abline(h = 0)
> legend("topright", legend = paste("theta =", c(2, 1, 0.5)),
+        lty = 3:1)
```

Theta-logistic growth rate

We plot the growth rate (a.k.a. the production function) for the theta-logistic model with $\theta < 1$, $\theta = 1$, and $\theta > 1$ (Fig. 3.13b).

```
> thetaGR.out <- sapply(theta, function(th) {
+     r * N * (1 - (alpha * N)^th)
+ })
> matplot(N, thetaGR.out, type = "l", col = 1)
> abline(h = 0)
> legend("bottomleft", legend = paste("theta =", c(2, 1, 0.5)),
+     lty = 3:1)
```

We also add an example of growth with low θ, but higher r.

Theta-logistic dynamics

We solve numerically for N, and plot the dynamics for N with $\theta < 1$, $\theta = 1$, and $\theta > 1$ (Fig. 3.13c).

```
> prms <- c(r <- 0.75, alpha <- 0.01, theta = 1)
> thetaN <- sapply(theta, function(th) {
+     prms["theta"] <- th
+     ode(y = 1, t.s, thetalogistic, prms)[, 2]
+ })
> matplot(t.s, thetaN, type = "l")
> legend("topleft", legend = paste("theta =", c(2, 1, 0.5)),
+     lty = 3:1, bty = "n")
```

3.4 Maximum Sustained Yield

The classical model of harvesting fisheries and wildlife populations is based on a particular conceptualization of *maximum sustained yield* (*MSY*). Maximum sustained yield is historically defined as the population size where both population growth rate and harvest rate are at a maximum. For the logistic growth model, this is half the carrying capacity. We can solve this by finding the maximum of the production function. To do this we use calculus (differentiation) to find where the slope of the production function equals zero. Starting with the partial derivative with respect to N (eq. 3.28), we solve for zero.

$$0 = r - 2r\alpha N \tag{3.39}$$

$$N = \frac{r}{2r\alpha} = \frac{K}{2} \tag{3.40}$$

Similarly, we could determine this peak for the θ-logistic model as well.

$$\frac{\partial \dot{N}}{\partial N} = r - (\theta + 1)\, r\, (\alpha N)^{\theta} \tag{3.41}$$

$$0 = \frac{K}{(\theta + 1)^{1/\theta}} \tag{3.42}$$

When $\theta = 1$, this reduces to $K/2$, as for the logistic model. As we saw above, $\theta < 1$ causes this value to be less than $K/2$.

Even if we assume the logistic model is appropriate for a given population, harvests in practice seldom reduce populations to $K/2$, because often N and K vary with time and are difficult to estimate. Moreover, there are economic forces that often lead to over-harvesting. More contemporary models of population harvesting incorporate these concepts [182]. We should note that when more detailed information is available on age- or stage-specific survivorship and reproduction, age- or stage-structured models of harvesting are the preferred method (e.g., Chapter 2). For many populations, such as marine fisheries, however, data are often available only on stock size (N) and total catch (H). In some fisheries models, the total catch is fixed and does not vary with N. This leads to highly unstable population dynamics and fortunately is no longer practiced in most fisheries. Usually the total catch H is modeled as a function of N.

Let us assume that total catch or harvest, H, is a simple linear function of N; the harvest rate FN is usually described this way.[15] In this case, a maximum sustained yield, MSY, can be determined from the logistic growth-harvest model. This basic model is then

$$dN/dt = rN(1 - \frac{N}{K}) - FN \tag{3.43}$$

where $FN = H$ harvest rate, and F is the per capita fishing mortality rate. Here we assume a fixed per capita catch mortality F so that total harvest is $H = F \times N$. MSY occurs at $K/2$ (Fig. 3.14b). Therefore would like to know the value of F for which a new $N^* = K/2$ rather than K. We can determine the value of F for which this is true by finding when $\mathrm{d}N/\mathrm{d}t = 0$ and $N = K/2$.

$$0 = r\frac{K}{2}\left(1 - \frac{\frac{K}{2}}{K}\right) - F\frac{K}{2} \tag{3.44}$$

$$F = \frac{r}{2} \tag{3.45}$$

[15] We could, but will not, further describe this by $F = qE$, where q is the catchability coefficient for the fishery, and E is fishing effort in hours.

MSY *and harvesting (Fig. 3.14a)*

Here we illustrate the interaction harvesting at a rate associated with *MSY* for logistic growth. We set logistic model parameters, and first plot logistic growth without harvest.

```
> r <- 0.5; alpha <- 0.01; N <- 0:105
> plot(N, eval(pop.growth.rate), type = "l", ylim = c(-3, 15),
+     ylab = "dN/dt and FN")
> abline(h = 0)
```

We then calculate F based on our solution eq. 9.6, and plot the linear harvest function with an intercept of zero, and slope of F.

```
> F <- r/2
> abline(c(0, F), lty = 2)
```

Equilibrium solution for logistic growth with harvesting (Fig. 3.14b)

When we add harvesting at rate $F = r/2$, we get a new equilibrium. Here we illustrate this using the same parameters as above, but now using the entire function with both growth and harvest.

```
> pgr.H <- expression(r * N * (1 - alpha * N) - F * N)
> N <- 0:55
> plot(N, eval(pgr.H), type = "l", ylab = "dN/dt (growth - harvesting)")
> abline(h = 0)
```

This merely represents the new equilibrium (where $\mathrm{d}N/\mathrm{d}t$ crosses the x-axis) with a harvest of rate $F = r/2$.

If harvesting occurs at a rate of $F < r/2$, the slope of the harvest line is shallower (cf. Fig. 3.14a), and we refer to this as *under-harvesting*. If the rate is $F > r/2$, then the slope is steeper (cf. Fig. 3.14a), and we refer to this as *over-harvesting*. Roughgarden and Smith (1996) show that under-harvesting is probably the most ecologically stable, whereas over-harvesting is frequently thought to be the economically optimal approach.

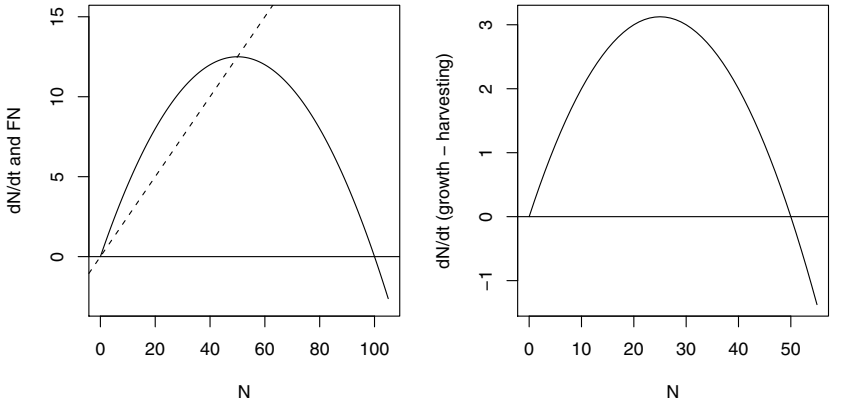
Typically the economically optimal approach leads to over-harvesting because the marginal income on a fishery is related to *the interest rate on invested income*, $F = (r + \rho)/2$, where ρ is the interest rate (usually ρ is set to 0.05). We can substitute this into eq. 3.43 and solve for N (ignoring $N = 0$).

$$0 = rN(1 - \frac{N}{K}) - \frac{r+\rho}{2}N \tag{3.46}$$

$$N\frac{r}{K} = r - \frac{r}{2} - \frac{\rho}{2} \tag{3.47}$$

$$N = \frac{K}{2}\left(1 - \frac{\rho}{r}\right). \tag{3.48}$$

Thus, unless $\rho = 0$, the typical economic optimum N_e^* will be smaller than $N* = K/2$. This makes intuitive economic sense if we consider income from invested



(a) Logistic Growth and Harvest Functions (b) Logistic Growth *with* Harvesting

Fig. 3.14: Illustration of maximum sustained yield under harvesting. (a) both the logistic growth model eq. 3.13 and the linear harvest model eq. 9.6, and (b) the combined growth model that includes harvesting as a source of mortality. Equilibria occur when the growth rate, y, equals zero, crossing the x axis.

profits. The margin of error, however, is much smaller, putting the fisheries population, and the investment of fishing fleets and associated econmoies at greater risk. Roughgarden and Smith [182] showed that ecological stability of a fishery is likely to beget economic and cultural stability. They show that the supposed "economic optimal" harvesting rate was so low that it increased the extinction risk of the harvested population to unacceptably high levels. The collapse of a fishery is, of course, both economically and culturally devastating.

3.5 Fitting Models to Data

Using theory to guide our data collection, and using data to guide our theory is, in part, how science progresses. In this section, we will use the use the theory of simple continuous logistic growth (linear density dependence) to frame our examination of the effects of nutrients on interactions of an alga that is embedded within a simple food web.

The focus of this section is the practice of describing data in terms of deterministic and stochastic models; this has been greatly facilitated by the R language and environment [13].

3.5.1 The role of resources in altering population interactions within a simple food web

Increasing resource availability may increase or, paradoxically, decrease the populations that use those resources [176,179,200]. Here we explore how adding re-

sources influences the strength of per capita denisty dependence, α, of an alga, *Closterium acerosum*, embedded in a simple food web [195].

The variety of observed responses to resource availability is sometimes mediated by the configuration of the food web in which a population is embedded. To explore the interaction between food web configuration and edibility, Stevens and Steiner [195] performed a simple experimental resource enrichment using a food web with one predator (*Colpidium striatum*) and two competing consumers which differed in their edibility; bacteria, which were edible and *Closterium acerosum*, which was inedible. Resources were added at two concentrations, a standard level, and a ten-fold dilution. We were interested in whether the inedible competitor *Closterium* could increase in abundance in the face of competition from bacteria. Although *Closterium* is inedible, it may be unable to increase in response to resource additions because it is competing with many bacterial taxa which appear to also be predator resistant. In addition, many bacteria are able to grow quickly in enriched environments. However, keystone predation theory [104] suggested that *Closterium* might prosper because the bacteria which coexist with the predator may be poor competitors, and the bacteria with the greatest capacity for utilizing higher resource levels may be kept in check by a predator which also might increase in abundance. Stevens and Steiner [195] presented time-averaged responses, perhaps related to K, because the predictions of keystone predation theory were focused on equilibrium predictions.

Here, we will explore the effects of the resource supply rate on two parameters of linear density dependence (logistic growth) r, and α of *Closterium acerosum*.

In the rest of this section, we will follow a generally cautious and pragmatic step-wise scheme which we adapt for our purposes here. We will

- import the data and examine its structure.
- look at the population dynamics of each replicate microcosm.
- consider whether we have outliers due to sampling.
- estimate parameters for one of the populations.
- fit logistic models to each population.
- fit a model to all populations and test the effects of nutrients on logistic growth parameters.

3.5.2 Initial data exploration

First we load two more libraries, the data, and check a summary.

```
> library(nlme)
> library(lattice)
> data(ClostExp)
> summary(ClostExp)

 Nutrients   No.per.ml           Day       rep          ID
 high:72   Min.   :   1.0   Min.   : 1.0   a:36   a.low  :18
 low :72   1st Qu.:  16.2   1st Qu.:11.0   b:36   d.low  :18
```

```
Median :  42.0   Median :26.0   c:36   c.low  :18
Mean   : 131.5   Mean   :25.9   d:36   b.low  :18
3rd Qu.: 141.5   3rd Qu.:39.0          c.high :18
Max.   :1799.1   Max.   :60.0          a.high :18
                                       (Other):36
```

Next we plot each replicate of the high and low nutrient treatments.

```
> xyplot(No.per.ml ~ Day | Nutrients, ClostExp, groups = rep,
+     type = "b", scales = list(relation = "free"), auto.key = list(columns = 4,
+         lines = T))
```

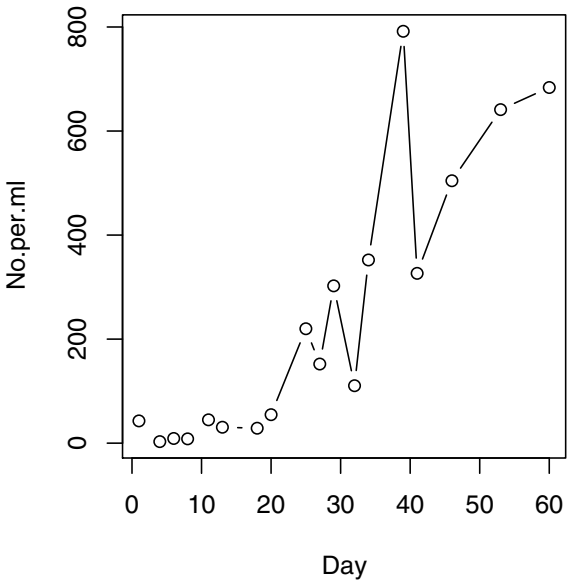


Fig. 3.15: Raw data for *Closterium* time series. Each line connects estimates of the *Closterium* popoulation in a single replicate microcosm. Each estimate is based on a ~ 0.3 mL sample drawn from the microcosm after it has been mixed. Low nutrient concentration is 1/10× the standard (high) concentration.

Right away, we might notice several things (Fig. 3.15). First, the high nutrient populations achieve higher abundances (note values on the *y*-axes). This seems consistent with the ten-fold difference in resource concentration [195]. Perhaps the second thing we notice is that the low nutrient populations seem more variable, relative to their own population sizes. Third, several observations seem remarkably high — which days?

```
> subset(ClostExp, Nutrients == "high" & No.per.ml > 1000)

Grouped Data: No.per.ml ~ Day | ID
    Nutrients No.per.ml Day rep     ID
```

```
114      high      1799  41   b b.high
116      high      1398  41   d d.high
```

```
> subset(ClostExp, Nutrients == "low" & No.per.ml > 100)
```

```
Grouped Data: No.per.ml ~ Day | ID
    Nutrients No.per.ml Day rep    ID
54        low     130.6  18   b b.low
55        low     144.0  18   c c.low
134       low     124.8  53   b b.low
142       low     158.2  60   b b.low
143       low     113.0  60   c c.low
```

What might we make of these observations? Are they mistakes, or important population dynamics, or some combination thereof? There may be several reasonable explanations for these data which we revisit at the end of this section. For now, we will assume that these points are no less accurate than the other values, and so will retain them.

Finally, we might note that the populations seem to decline before they increase (days 0–5, Fig. 3.15). These declines might well represent a bit of evolutionary change, under the new environment into which these populations have been introduced [56] from their stock cultures; we might consider removing the first two sample points to avoid confound mechanisms of evolution versus simple growth. In general, however, we should be cautious about throwing out data that do not conform to our expectations — let us leave these data in, for the time being.

Now we estimate the parameters of deterministic logistic growth, first without, and then with, an explicit consideration of time.

3.5.3 A time-implicit approach

How might we estimate the parameters without explicitly considering the temporal dynamic? Recall the relation between per capita population growth *vs.* population size,

$$\frac{\mathrm{d}N}{N\mathrm{d}t} = r - r\alpha N. \tag{3.49}$$

Here we see that per capita growth rate is a linear function (i.e., $y = mx + b$) of population density, N. Therefore, all we have to do is calculate per capita growth rate from our data and fit a simple linear regression of those growth rates versus N. The y-intercept (when $N = 0$) will be our estimate of r, and the slope will be an estimate of $r\alpha$.

We will estimate per capita growth rate, *pgr*, with

$$pgr = \frac{\log\left(N_{t+i}/N_t\right)}{i} \tag{3.50}$$

where i is the time interval between the two population densities [188]. In this and other contexts, logarithm of two populations is sometimes referred to simply as the *log ratio*.

Let's estimate this for one of the microcosms, high nutrient, replicate c.

```
> Hi.c <- subset(ClostExp, Nutrients == "high" & rep == "c")
```

We calculate the differences in density, the time intervals over which these changes occured, and the per capita growth rate.

```
> n <- nrow(Hi.c)
> N.change <- Hi.c$No.per.ml[-1]/Hi.c$No.per.ml[-n]
> interval <- diff(Hi.c$Day)
> pgr <- log(N.change)/interval
```

We can then (i) plot these as functions of N_t (as opposed to N_{t+1}), (ii) fit a linear regression model through those points, and (iii) plot that line (Fig. 3.18a).

```
> Nt <- Hi.c$No.per.ml[-n]
> plot(pgr ~ Nt)
> mod1 <- lm(pgr ~ Nt)
> abline(mod1)
```

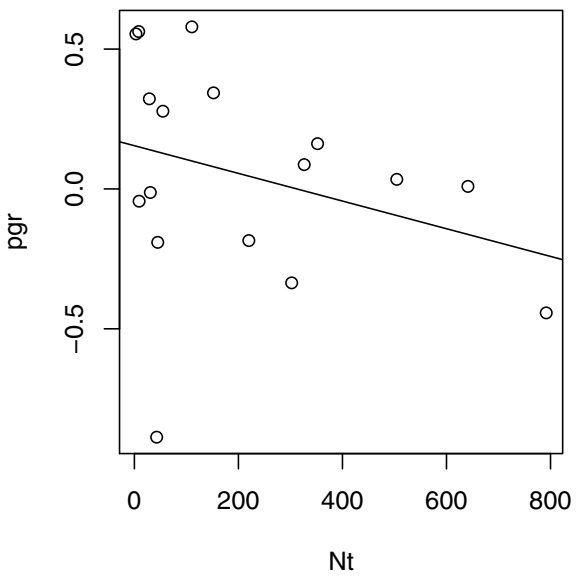


Fig. 3.16: Per capita growth rate *vs.* population size at the beginning of the interval (data from one high nutrient replciate).

The linear regression model defines a best fit straight line through these points,

```
> summary(mod1)

Call:
lm(formula = pgr ~ Nt)

Residuals:
```

```
   Min     1Q Median     3Q    Max
-1.021 -0.206  0.129  0.182  0.480

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.154556   0.125327    1.23     0.24
Nt          -0.000495   0.000395   -1.25     0.23

Residual standard error: 0.382 on 15 degrees of freedom
Multiple R-squared: 0.0944,        Adjusted R-squared: 0.0341
F-statistic: 1.56 on 1 and 15 DF,  p-value: 0.23
```

where y-intercept is the first coefficient of the linear regression; this is an estimate of r (eq. 3.49). The slope of the line is the second coefficient and is $r\alpha$ (eq. 3.49).

Now we can do this with all the replicates. Rearranging the data is a bit complicated, but this will suffice. Essentially, we split up the `ClostExp` data frame into subsets defined by nutrient and replicate ID, and then apply a function that performs all of the steps we used above for a single microcosm. The output will be a *list* where each component is a new small data frame for each microcosm.

```
> EachPop <- lapply(split(ClostExp, list(ClostExp$Nutrients,
+     ClostExp$rep)), function(X) {
+     n <- nrow(X)
+     N.change <- (X$No.per.ml[-1]/X$No.per.ml[-n])
+     interval <- diff(X$Day)
+     data.frame(Nutrients = as.factor(X$Nutrients[-n]),
+         rep = as.factor(X$rep[-n]),
+         pgr = log(N.change)/interval, Nt = X$No.per.ml[-n])
+ })
```

Next we just stack all those individual data frames up into one.

```
> AllPops <- NULL
> for (i in 1:length(EachPop)) AllPops <- rbind(AllPops, EachPop[[i]])
```

Finally, we plot all of the microcosms separately.

```
> xyplot(pgr ~ Nt | rep * Nutrients, AllPops, layout = c(4,
+     2, 1), scales = list(x = list(rot = 90)), panel = function(x,
+     y) {
+     panel.grid(h = -1, v = -4)
+     panel.xyplot(x, y, type = c("p", "r"))
+ })
```

The biggest difference about these populations appears to be that the carrying capacities differ between the high and low nutrients (Fig. 3.17). First note that the y-intercepts are all remarkably similar, indicating that their r are all very similar. The big differences between the high and low nutrients are that the slopes of the relations and the x-intercepts are both very different. The slope is $r\alpha$, while the x-intercept is K. Next we fit a statistical model to these data.

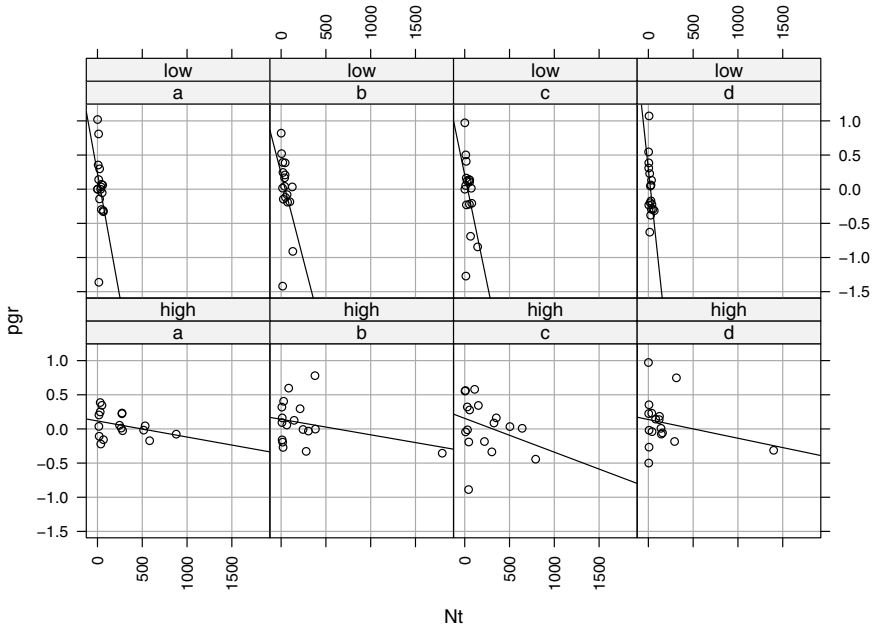


Fig. 3.17: Per capita growth rates *vs.* population size, N, at the beginning of the interval for which the growth rate is calculated (includes all microcosms). Note that y-intercepts are all fairly similar, while the maximum N, slopes and x-intercepts differ markedly.

A statistical model of per capita growth

A *mixed model*[16] is a statistical model that includes both random effects (e.g., blocks, or subjects) and fixed effects (e.g., a treatment) [161]. Let's not worry about the details right now, but we'll just use it to test for differences in slope, where we have measure the same microcosms repeatedly through time.

Previously, we showed that it makes at least a modicum of sense to fit eq. 3.49 to each microcosm (Fig. 3.17). Having done that, we next fit all of these to a single mixed model, in which the microcosm is our *block* or *subject* which may exert some unknown random effect on the populations. First we make sure we have a unique ID label for each microcosm.

```
> AllPops$ID <- with(AllPops, Nutrients:rep)
> modSlope <- lme(pgr ~ Nt + Nutrients + Nt:Nutrients, data = AllPops,
+     random = ~1 | ID)
```

The above code specifies that per capita growth rate, `pgr`, depends in part upon population size, `Nt` — this will be the slope of the `pgr Nt` relation. The effect of

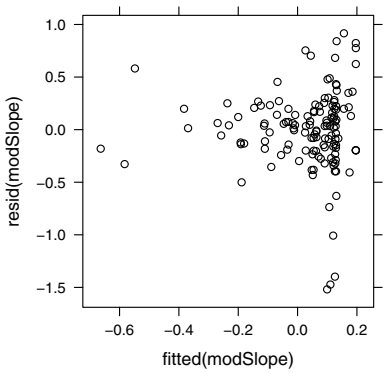
[16] This section is advanced material. You may want to skim over the parts you don't understand, and focus on conclusions, or perhaps the general process.

nutrient level, `Nutrients`, will simply alter the intercept, or height, of the lines. The code also specifies that the slope the relation can depend upon the nutrient level, `Nt:Nutrients`. Finally, the code also specifies that each microcosm makes its own unique and *random* contribution to the per capita growth rate of its *Closterium* population.

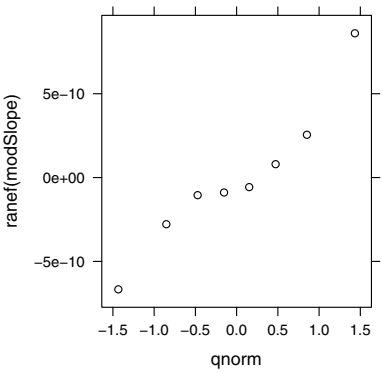
In a proper statistical analysis, it would be necessary to first check the assumptions of this model, that the noise (the residuals) and the random effects are both normally distributed. A plot of the residuals *vs*. the fitted or predicted values should reveal a scatter of points with relatively constant upper and lower bounds.

```
> xyplot(resid(modSlope) ~ fitted(modSlope))
```

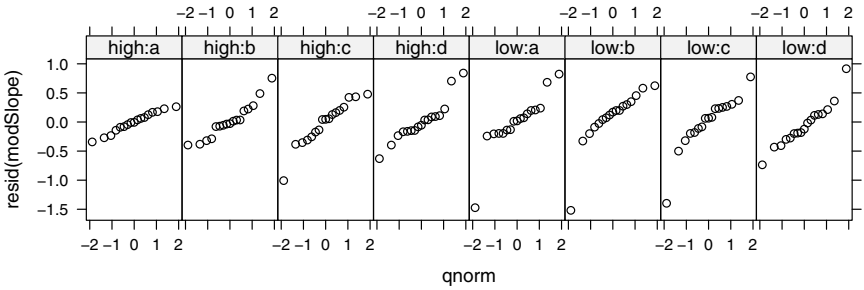
Our plot (Fig. 3.18a) reveals larger residuals for larger fitted values — this is not good. Quantile-quantile plots of the residuals and random effects should



(a) Residuals *vs*. Fitted



(b) Random Effects Q-Q Plot



(c) Residuals Q-Q Plot

Fig. 3.18: Diagnostic plots for fitting the per capita growth rate to population size.

reveal points along a straight diagonal line.

```
> qqmath(~resid(modSlope) | ID, data = AllPops, layout = c(8,
+     1, 1))

> qqmath(~ranef(modSlope))
```

Our random effects (Fig. 3.18b) are very small, indicating only small systematic differences among microcosms that are not accounted for by the treatment. Nonetheless, their scatter of points along the diagonal are fairly consistent with a normal distribution. The residuals (Fig. 3.18c) for each microcosm seem fairly normally distributed except that several populations have outliers (replicates low:a-c, high:c). Taken as a group (Fig. 3.18a), we see that the residual variation increases the mean (fitted). This is at least in part due to having more observations at larger values, but is also a typical and common problem — variables with small absolute values simple cannot vary as much as variables with large absolute value. Therefore, we specify that that model takes that into account by weighting residuals observations differently, depending upon the expected (i.e. fitted or mean) value.

```
> modSlope2 <- update(modSlope, weights = varExp())
```

We then compare these two models,

```
> anova(modSlope, modSlope2)

          Model df   AIC   BIC logLik   Test L.Ratio p-value
modSlope      1  6 169.1 186.4 -78.55
modSlope2     2  7 163.9 184.1 -74.97 1 vs 2   7.153  0.0075
```

and find that the slightly more complicated one (`modSlope2` with 7 degrees of freedom) is significantly better (likelihood ratio = 7.1, $P < 0.05$) and is more parsimonious (smaller AIC) [18].

Now we provide a preliminary test of the fixed effects (nutrients and population size) with analysis of variance.

```
> anova(modSlope2)

             numDF denDF F-value p-value
(Intercept)      1   126   0.823  0.3659
Nt               1   126   3.665  0.0578
Nutrients        1     6  11.198  0.0155
Nt:Nutrients     1   126  27.659  <.0001
```

The first thing we see in this *analysis of variance* output is that the effect of population size, N_t, depends upon nutrient level (`Nt:Nturients`, $P < 0.0001$). Given this, we should avoid drawing conclusions based on the main effects of nutrients. Next we go on to estimate intercepts and slopes associated with different treatments.

```
> summary(modSlope2)$tTable
```

```
                    Value Std.Error  DF t-value   p-value
(Intercept)      0.1336241 0.0568611 126  2.3500 2.033e-02
Nt              -0.0002861 0.0001146 126 -2.4965 1.383e-02
Nutrientslow     0.0641174 0.0878452   6  0.7299 4.930e-01
Nt:Nutrientslow -0.0056023 0.0010652 126 -5.2592 6.014e-07
```

Given R's default contrasts (linear model parameters), this output indicates that the intercept (our estimate of r) for the high nutrient group is ~ 0.13, and the slope of the high nutrient microcosms is -0.000286. The intercept for the low nutrient level is calculated as $0.13 - 0.07$, and we find it is not significantly smaller ($P > 0.05$). On the other hand, the slope of the low nutrient microcosms is $-0.000278 - 0.00573$, which is significantly smaller than the slope of the high nutrient microcosms ($P < 0.05$). This information can guide another take on these data.

```
> cfs <- fixef(modSlope2)
> cfs

    (Intercept)              Nt    Nutrientslow Nt:Nutrientslow
      0.1336241      -0.0002861       0.0641174      -0.0056023

> -cfs[2]/cfs[1]

      Nt
0.002141

> -(cfs[2] + cfs[4])/(cfs[1] + cfs[3])

     Nt
0.02978
```

indicating that the per capita effects are much smaller in the high nutrient treatment. (The label `Nt` is an unimportant artifact, retained from using the coefficients to calculate the values.) Next, we use these estimates to help us fit a times series model.

3.5.4 A time-explicit approach

Here we use the above information to help us fit the time-explicit version of the logistic growth model. In this section, we

1. create and examine a nonlinear logistic model,
2. fit the model to each microcosm using `nlsList` to confirm it makes sense, and
3. fit a single nonlinear mixed model that tests which logistic growth parameters varies between treatments.

At each step, we examine the output and diagnose the fit; it should give you a very brief introduction to this process. For more thorough treatments, see [13, 161].

First, we code in R a model of eq. 3.25, that is, logistic growth with parameters α, r, and N_0.

```
> ilogistic <- function(t, alpha, N0, r) {
+     N0 * exp(r * t)/(1 + alpha * N0 * (exp(r * t) - 1))
+ }
```

Let's make sure this makes sense by plotting this curve on top of the raw data, focusing on the raw data and model parameters for the high nutrient microcosms.

```
> plot(No.per.ml ~ Day, ClostExp, subset = Nutrients == "high")
> curve(ilogistic(x, alpha = -cfs[2]/cfs[1], N0 = 6, r = cfs[1]),
+     1, 60, add = T, lty = 2)
```

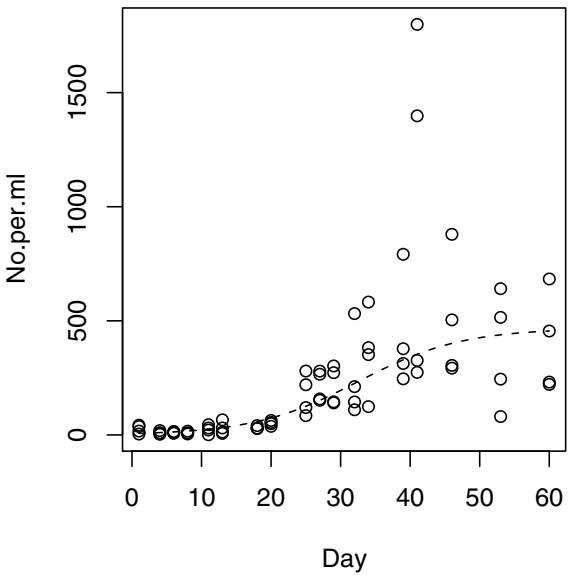


Fig. 3.19: High nutrient microcosms and the logistic growth model using coefficients from `modSlope`.

Our model and the coefficients we generated above seem to be a reasonable approximation of the data (Fig. 3.19). Now let's use `nlsList` to fit the logistic model to each microcosm separately.

```
> Cmod.list <- nlsList(No.per.ml ~ ilogistic(Day, alpha, N0,
+     r) | ID, data = ClostExp, start = c(alpha = 1/100, N0 = 5,
+     r = 0.15), control = list(maxiter = 1000))
```

We note that models cannot be fit for three microcosms. For now, we plot the coefficients for each microcosm for which a model was fit.

```
> plot(coef(Cmod.list), layout = c(3, 1, 1))
```

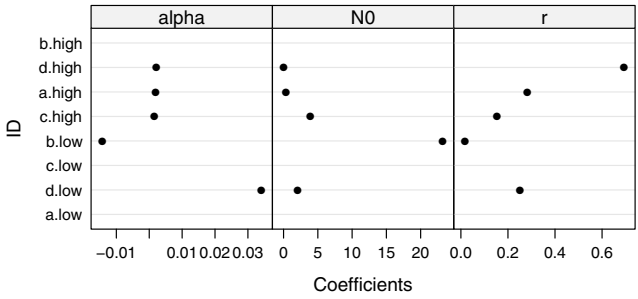


Fig. 3.20: Coefficients estimated for each microcosm separately. Missing coefficients occurred when model fits could not be easily acheived.

We can see that the coefficients (Fig. 3.20) are somewhat similar to those we estimated above with `modSlope2`. We note, however, that for one replicate $\alpha < 0$, which does not make sense. If we check the residuals,

```
> plot(Cmod.list)
```

we see that the residuals increase dramatically with the fitted values (graph not shown). This is consistent with our previous observations.

In this next step, we create a single nonlinear mixed model using `nlme`. We use `pdDiag` to specify that each microcosm has its own unique effect on each of the parameters, aside from nutrient concentrations. We use `weights` to specify that the variance of the residuals is a power function (a^{2t}) of the mean, and which fits an extra parameter, t.

```
> Cmod.all <- nlme(Cmod.list, random = pdDiag(form = alpha +
+     N0 + r ~ 1), weights = varPower())
```

We saw above that the nutrients caused a big difference in the slopes of the *pgr vs. N* relations. Now we would like to estimate the *alpha* for each level of nutrients. In estimating these, we can also test whether they are different.

First we capture the fixed effect coefficients of the current model.

```
> cfs2 <- fixef(Cmod.all)
> cfs2

   alpha       N0        r
0.009266 7.143118 0.130551
```

Next we update the model, specifying that `alpha` should depend on nutrient level, that is, estimate one α for each level of nutrient. We specify that the other paramters are the same in both treatments. This will mean that we have one additional coefficent, and we need to provide the right number of starting values, and in the correct order (`alpha1, alpha2, N0, r`).

```
> Cmod.all2 <- update(Cmod.all, fixed = list(alpha ~ Nutrients,
+     N0 + r ~ 1), start = c(cfs2[1], 0, cfs2[2], cfs2[3]))
```

It would be good practice to plot diagnostics again, as we did above. Here, however, we push on and check the confidence intervals for our parametersa summary.

```
> summary(Cmod.all2)

Nonlinear mixed-effects model fit by maximum likelihood
  Model: No.per.ml ~ ilogistic(Day, alpha, N0, r)
 Data: ClostExp
   AIC  BIC logLik
  1549 1575 -765.3

Random effects:
 Formula: list(alpha ~ 1, N0 ~ 1, r ~ 1)
 Level: ID
 Structure: Diagonal
        alpha.(Intercept)    N0        r Residual
StdDev:         6.533e-20 1.392 2.56e-09    1.488

Variance function:
 Structure: Power of variance covariate
 Formula: ~fitted(.)
 Parameter estimates:
 power
0.8623
Fixed effects: list(alpha ~ Nutrients, N0 + r ~ 1)
                   Value Std.Error  DF t-value p-value
alpha.(Intercept)  0.002    0.0003 133   5.021       0
alpha.Nutrientslow 0.016    0.0026 133   6.185       0
N0                 6.060    1.4245 133   4.254       0
r                  0.139    0.0144 133   9.630       0
 Correlation:
                   al.(I) alph.N N0
alpha.Nutrientslow -0.019
N0                 -0.289  0.048
r                   0.564  0.058 -0.744

Standardized Within-Group Residuals:
    Min      Q1     Med      Q3     Max
-1.3852 -0.6010 -0.2038  0.3340  4.5236

Number of Observations: 144
Number of Groups: 8
```

If we look first at the random effects on the parameters, we see that the variance components for `alpha` or `r` are vanishingly small, we should therefore remove them from the analysis, and then compare the two models.

```
> Cmod.all3 <- update(Cmod.all2, random = N0 ~ 1)
> anova(Cmod.all2, Cmod.all3)
```

```
          Model df  AIC  BIC logLik   Test   L.Ratio p-value
Cmod.all2     1  9 1548 1575 -765.3
Cmod.all3     2  7 1544 1565 -765.3 1 vs 2 0.0008463  0.9996
```

We see that the less complex model is not significantly different ($P > 0.05$) and is more parsimonious (smaller AIC, Akaike's information criterion, [18]).

At last, we can examine the confidence intervals to get the estimates on the intraspecific competition coefficents, α_{ii}.

```
> intervals(Cmod.all3)$fixed

                       lower     est.   upper
alpha.(Intercept)  0.000978 0.001599 0.00222
alpha.Nutrientslow 0.011095 0.016206 0.02132
N0                 3.279937 6.057370 8.83480
r                  0.110739 0.138860 0.16698
attr(,"label")
[1] "Fixed effects:"
```

see that the analysis, such as it is, indicates that the confidence intervals for the estimated α do not overlap, indicating that the expected values are very, very different from each other. We could calculate the bounds on K_{high} and K_{low} as well, based on our lower and upper confidence limits on α_{high}, α_{low}.

```
> 1/intervals(Cmod.all3)$fixed[1:2, c(1, 3)]

                     lower  upper
alpha.(Intercept)  1022.41 450.43
alpha.Nutrientslow   90.13  46.91
```

This aproximately ten-fold difference in K seems surprisingly consistent with the ten-fold difference in nutrient concentration initially established in the experiment. This ten-fold increase in *Closterium* may be merely an odd coincidence because it implies that all species in the food web should have increased ten-fold, but that was not the case [195]. On the other hand, it may guide future experiments in keystone predation.

We can plot the fixed effects, and also the added variation due random variation in N_0 and r due to the ID of the microcosms.

```
> plot(augPred(Cmod.all3, level = 0:1), layout = c(4, 2, 1),
+     scales = list(y = list(relation = "free")))
```

When we plot our data along the with the model (Fig. 3.21), we can see the disparity between the predictions and the data — the source of those big residuals we've been ignoring.

There may be at least a couple issues to consider when we think about the disparities.

Sampling error We do not have much idea about how accurate or precise our sampling is; this might be remedied in the next experiment by using a mean with multiple samples per time interval.

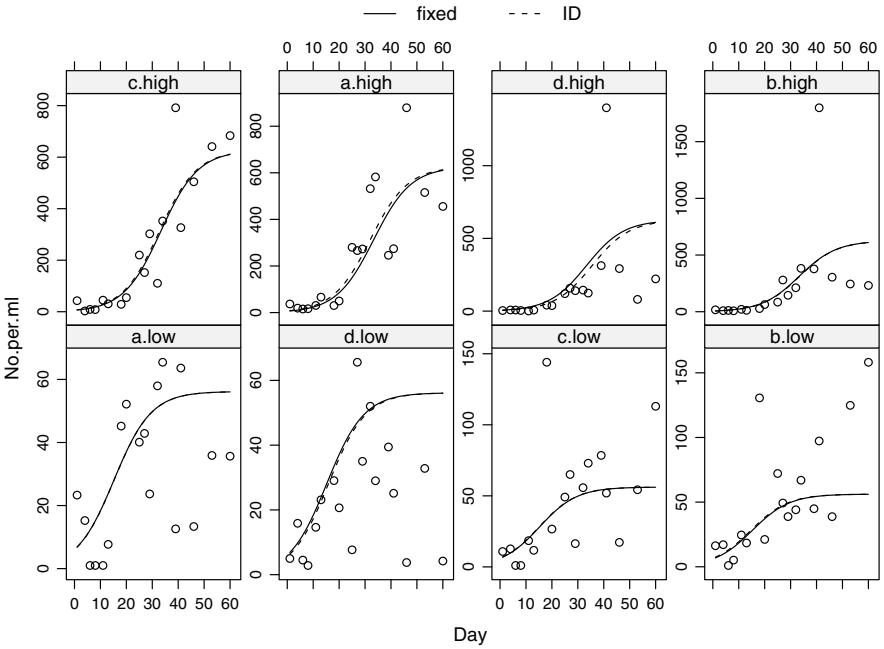


Fig. 3.21: Times series plots augmented with the overall predictions (solid lines for the fixed effects) and predictions for individual microcosms (dashed lines).

Experimental Design In this system, 10% of the medium was replaced each week (to remove waste products and replenish nutrients), and this is likely to introduce variability.

Dynamics Species interactions may drive dynamics, derived from feedback between (i) bacteria recycling nutrients, (ii) bacteria consuming nutrients, (iii) the predator *Colpidium* consuming bacteria, and (iv) *Closterium* consuming nutrients (Chapters 5, 6).

Perhaps the first steps for an improved experiment might include sampling at least twice for each sample period, to at least find out what our sampling error is. It may, or may not, be a problem, but a few trials would let us know. We also might increase the frequency and decrease the magnitude of the medium replacement, and to make sure the timing of the sampling corresponds appropriately with the medium replace. After we take care of these issues, then we can take a harder look at the dynamics.

Conclusions

Our results show that lowering nutrient concentration increased the intraspecific interaction strengths, α_{ii}, ten-fold. Alternatively, we could express this as a ten-fold change in the carrying capacity K of these populations. The effect of

resources on interaction strength is sometimes a heated topic of discussion, and our precise definitions, derived from simple growth models, provides an unambiguous result. Our analysis has also pointed to ways that we might perform better experiments. More generally, by framing our experiment in terms of existing theory, we can interpret our results in a more precise manner, and facilitate better advances in the future [13]. This will help us make more constructive contributions to the field.

3.6 Summary

In this chapter, we motivated density dependence, focusing specifically on discrete and continuous logistic growth which assumes linear density dependence. We explored how algebraic rearrangements can yield useful and interesting insights, and how we might think about generalizing or simplifying logistic growth. We showed how the time lag built into the discrete model results in chaos. We explored ideas of maximum sustained yield, and how we might fit the continuous logistic growth model to real data.

Problems

3.1. Dynamics of an annual plant
(a) Calculate r_d of an annual plant that has a maximum growth rate of $N_{t+1}/N_t = 2$ at very, very small population sizes.
(b) Calculate the appropriate per capita density dependent effect of an annual plant with a carrying capacity K of 100 inds·m^{-2}.
(c) Write the appropriate logistic growth equation that incorporates the intrinsic growth of (a) and the density dependence of (b).
(d) Graph the 10 y dynamics ($t = 0, \ldots, 10$) of the annual plant in (a) and (b), starting with $N_0 = 1$.

3.2. Dynamics of *E. coli*
(a) Calculate r of *E. coli* that has a doubling time of 30 min. Express this rate in hours.
(b) Calculate the per capita density dependent effect of an *E. coli* culture that grows logistically over a 24 h period, and which levels off at a density of 10^7 CFU·mL^{-1} (CFU is colony forming units — for *E. coli* its is equivalent to individuals).
(c) Graph the 50 h dynamics ($t = 0, \ldots, 50$) of the *E. coli* population in (a) and (b), starting with $N_0 = 1000$.

3.3. Nonlinear Density Dependence
Sibly et al. [188] found that most species have nonlinear and concave-up density dependence. They use the θ-logistic growth model. (a) Create a theta-logistic

continuous growth model for use with the `ode()` function in the `deSolve` package.
(b) Show that with $\theta = 1$, it is identical our function `clogistic` above.
(c) Graph N for $t = 0, \ldots, 100$ using a few different values of θ and explain how θ changes logistic growth.

3.4. Harvested Populations
The logistic growth equation and other similar equations have been used and abused in efforts to achieve a *maximum sustained yield* of a harvested population. The immediate goal of maximum sustained yield management practices is to kill only the number of individuals that reduces the population to half of its carrying capacity, assuming that eq. 3.13 describes the population growth. Answer the questions below to help you explain why this would be a goal.
(a) Find expressions for population growth rates when $N = K/4$, $K/2\,3K/4$ (substitute these values for N in eq. 3.13, and show your work). Which of these results in the highest growth rate? How do these relate to the management of a harvestetd population?
(b) Show the derivation of the partial derivative of the continuous logistic growth model, with respect to N (i.e., $\partial\dot{N}/\partial N$). Solve for zero to determine when total population growth rate reaches a maximum. Show your work.
(c) What would be the ecological and economic rationale for *not* killing more individuals, and keeping $N > K/2$?
(d) What would the consequences be for the population if you *assume* linear density dependence $(1 - \alpha/N)$, but in fact the population is governed by nonlinear density dependence where $\theta < 1$ and *theta* > 1 (Figs. 3.13a-3.13c)?
(e) What economic benefit would you gain if you harvested the entire population all at once (and eliminated it from the face of the planet)? What could you do with all that money?
(f) How would you incorporate both harvesting and economic considerations into your logistic growth model?

3.5. Environmental Variability
Most environments change continually. Temperature, resource availability, changes in predator or pathogen abundance all influence the carrying capacity of the environment.
(a) Use the discrete version of the logistic growth equation to model a population in a variable environment. Do this by creating a discrete logistic growth function that adds (or subtracts) a random amount to K in each time step. Use one of the many functions that can draw random numbers from particular distributions (e.g., `rpois()`, `rlnorm()`, `runif()`). You might start by playing with one of the random number generators:

```
Kstart <- 100; time <- 1:20; K <- numeric(20);
for(i in 1:20) K[i] <- Kstart + rnorm(1, m=0, sd=5);
plot(time, K).
```

(b) All distributions are characterized by their *moments*. For instance, the Normal distribution is typically defined by its mean, μ, and variance, σ^2. Focus on just one moment of your selected distribution in (a), and use your simulations

to determine quantitatively the relation between K and the resulting N derived from the discrete growth model. For instance, you might vary the standard deviation of the random normal distribution that contributes to K, and examine how the standard deviation of K, σ_K relates to mean N, μ_N.
(c) Create a reddened time series for K.[17](Hint: What are the values of x and y when you do `x <- sample(c(-1,0,1), 20, replace=TRUE); y <- cumsum(x)` ?). Use this time series to create a reddened population dynamic. (Hint: First create the vector of reddened K's equal in length to your time series. Then create a growth function that can access the vector, e.g. `DLG.RK <- function(alpha=0.001, rd=1, N0=10, gens=20, K=K.red)`. Then, inside the for loop, use an indexed K, such as `1-N[t]/K[t]`.

[17] Environmental factors such as temperature frequently vary in a gradual fashion, such that if the weather is hot today, it is likely to be hot tomorrow. Such variation is described in terms of auto-correlation or a spectral distribution or color [67,155]. Spectral decomposition of a times series involves a description of the series in terms of waves of different wavelengths or frequencies. White noise is variation that is described by equal contributions of all wavelengths (hence the term "white"), and it is a series that is completely random from one time step to the next. Reddened environmental variation is variation that is described by a predominance of longer wavelengths (hence the term "red"), and it is a series in which values tend to change gradually, where values that are close together in time tend to be more similar, or auto-correlated. Spectral variation is also referred to as $1/f$ noise ("one over f noise").