Simple Population Models

Overview: *Dynamic population models track the size of a population with the passage of time. The simplest population models in ecology make several assumptions: Individuals are assumed to be genetically and behaviourally identical, they are fully mixed, they experience the same conditions from their environment and they are not subject to any form of randomness. Nevertheless, even these simple models can display surprisingly complex dynamics. We examine two such models in discrete time one without and one with density dependence.*

1. The Study System

Red Grouse (*Lagopus lagopus scoticus*) is a medium-sized bird of the grouse family, a subspecies of the circumpolar willow grouse. It is endemic to the British Isles and its range is specific to Scotland, northern England and Ireland where it is managed for game shooting. Red grouse are herbivorous, feeding mainly on the shoots, seeds and flowers of heather and, also, on berries, cereal crops and sometimes insects. Males are territorial and monogamous. Mating pairs begin to form during the autumn and males become increasingly territorial as winter progresses. The majority of mortality occurs during winter, and annually, a maximum of only 50% of adult birds survive to breed. In spring, the pairs set up their nest, a shallow scrape up to 20 cm across which is lined with vegetation. About six to nine eggs are laid, mainly during April and May. They are oval, glossy and pale yellow with dark brown blotches. The eggs are incubated for 19 to 25 days, the chicks can fly 12 to 13 days after hatching and are fully grown after 30 to 35 days. Under ideal environmental conditions, an average of 3.4 eggs survive through spring and summer to adulthood. In autumn, young cocks compete for territories along with the adult survivors from the previous year. The main limiting factor to population growth in red grouse is habitat availability. Males are fiercely territorial and this restricts the number of breeding pairs that can fit in a given moor each spring.

Red grouse cock

2. Ecological Theory

In populations with discrete breeding seasons, the intrinsic growth rate of a population is the net, per-capita change in a single time unit. This includes all forces of population change; deaths, births and migration. If it is greater than 0, then an unrestricted population will increase indefinitely. If it is negative, then the population will decline towards extinction. However, unrestricted population growth is unsustainable, as was poignantly expressed by Thomas Malthus in the early 1800s:

"The power of population is so superior to the power of the earth to produce subsistence for man, that premature death must in some shape or other visit the human race. The vices of mankind are active and able ministers of depopulation. They are the precursors in the great army of destruction, and often finish the dreadful work themselves. But should they fail in this war of extermination, sickly seasons, epidemics, pestilence, and plague advance in terrific array, and sweep off their thousands and tens of thousands. Should success be still incomplete, gigantic inevitable famine stalks in the rear, and with one mighty blow levels the population with the food of the world."

3. Mathematical Theory

We wish to model population size (P_t) with passing years (*t*). In a sexually reproducing population, the **birth rate** is the number of new offspring per adult animal (male and female). Given a 1:1 sex ratio, the number of females in year *t* is $P_t/2$. The **recruitment rate** is the number of adults in the next generation produced by each adult animal in the current generation. The recursive equation for this biological system is

$$P_{t+1} = P_t + \text{recruitment} - \text{adult deaths} \tag{1}$$

Let us now specify the timing of events during a single year. Egg laying occurs during a narrow time interval in spring. We take this to be the beginning of the year. In contrast, mortality (of both chicks or adults) occurs continuously throughout the year. Total adult mortality can be expressed as the product of per-capita mortality and adult population size at the beginning of the year. The eggs laid in spring will eventually yield the population's next batch of recruits. Therefore, by using recruitments rather than births, chick mortality is incorporated implicitly.

  If we use *d* to denote the proportion of adults dying during the year and *b* the number of offspring per parent that reach adulthood, then the model can be written

$$\begin{aligned} P_{t+1} &= P_t + bP_t - dP_t \\ &= (1 + b - d)P_t \end{aligned} \tag{2}$$

It is customary to simplify notation by replacing the difference $b - d$ by a single number *r* called the population's **intrinsic growth rate**.

$$P_{t+1} = (1 + r)P_t \tag{3}$$

If the intrinsic growth rate is zero, then $P_{t+1} = (1 + 0)P_t = P_t$, meaning that each individual in the population this year is replaced by exactly one individual in the next year. This may either be the same animal, its own offspring, or somebody else's. If *r* is positive (recruitment exceeds mortality), say 0.1, then the population next year is multiplied by 1.1 so it is 110% of what it was in the previous year. If *r* is negative (mortality exceeds recruitment), say -0.1, then the population declines by 10% each year. The minimum value of the intrinsic growth rate ($r = -1$) is achieved when no adults survive ($d = 1$) and no offspring make it to adulthood ($b = 0$). This lower bound for *r* is realistic because a value smaller than -1 in eq. (3) would lead to negative population sizes.

  Would you feel comfortable using eq. (3) to describe a population of real animals? Its main problem is that it is a geometric progression and therefore predicts that, if $r > 0$, the population will increase indefinitely. That is not true of real populations whose size is ultimately restricted by limited resources such as prey, nutrients and refuge sites. In the case of grouse, territoriality and habitat availability restrict the number of breeding pairs each year.

  When population growth rate changes with population size (for example due to competition for limited resources) it is said to be **density dependent**. To model this phenomenon, we assume that the population's environment can support a limited and fixed number of animals. We call this the environment's **carrying capacity** and symbolise it by *K*. As the population approaches this size, it will grow more slowly. If the population's size exceeds the environment's carrying capacity then it will start to decrease. We therefore need to model the intrinsic growth rate as a decreasing function of population size. The simplest possible formulation is a linear function that crosses zero at *K*,

$$r = r_{\max}\left(1 - \frac{P_t}{K}\right) \tag{4}$$

The constant $r_{\max}$ is the growth rate observed when the population is very small relative to the carrying capacity. To check that it does what it is supposed to, examine eq. (4) under the following four scenarios i) $P_t = 0$, ii) $P_t < K$, iii) $P_t = K$, iv) $P_t > K$. Placing this back into the population model yields what is known as the **discrete logistic population model**

$$P_{t+1} = P_t + r_{\max}\left(1 - \frac{P_t}{K}\right)P_t \qquad (5)$$

4. R Ingredients

■ *Set-indexing and element extraction:* Consider the set of numbers `A`, defined as `A<-c(2,4,1,6,0,5)`. The documentation of R refers to such sets as **vectors.** The R-equivalent of A_n (the nth element of the set A) is `A[n]`. In the example above, if you type `A[1]`, R will return `2`, typing `A[3]` will give you `1`. In addition to this syntax, there are several other ways of using indexing to get specific subsets of a set:

If you want…	…the R-syntax is	Example input	Example result
The nth element of a set A	`A[n]`	`A[5]`	0
All but the nth element of a set A	`A[-n]`	`A[-5]`	2 4 1 6 5
The first n elements	`A[1:n]`	`A[1:5]`	2 4 1 6 0
The last n elements	`A[-(1:n)]`	`A[-(1:3)]`	6 0 5
Specific elements from A	`A[c()]`	`A[c(1,3)]`	2 1
Elements below (or above) a value a	`A[A>a]`	`A[A>2]`	4 6 5
Elements between a and b (if $a<b$)	`A[A>a & A<b]`	`A[A>2 & A<6]`	4 5
Elements between a and b, inclusive	`A[A>=a & A<=b]`	`A[A>=2 & A<=6]`	2 4 6 5

■ *Iterating the general definition of a sequence:* To generate the first 21 terms of the arithmetic progression $a_{t+1} = 100 + 35a_t$ we can first create an index `n` and then apply to it the general definition of an arithmetic progression to obtain the sequence `a`.

```
n<-0:20
a<-100+n*35
```

Note that `n` is a list of integers and so is `a`. The same result can be achieved by merging the two lines of code

```
a<-100+(0:20)*35
```

If we want to retrieve the 10th term of this sequence, we need to type a[11]. This is because our index `n` includes the value 0 but lists in R are only ever indexed from 1 upwards.

■ *Plotting time series:* Given a list of values (say, `list`) representing the time series you can simply ask R to `plot(list)`. The options, `xlab="time"` and `ylab="Population"` will put x and y labels on the plot. If you want the dots to be joined, you can use the option `type="l"`. If you want to colour your curve red, add the option `col="Red"`. If you want to superimpose the plot of another dataset (say, `list2`) to the existing one, type `lines(list2)`. We will deal more extensively with graphs in future practicals.

■ *Iteration and loops:* Sequences are ideal for introducing the concept of iteration, i.e. the repeated application of a set of commands. If the results of one iteration do not depend on the results of the previous ones this can be done easily as we saw above. However, difference equations, are recursive. This requires a different programming device, called a **loop**, which comprises of 1) a loop declaration and 2) the main body of the loop. In R, a typical loop will look a bit like this:

```
for (counter in min:max) #  This line declares the loop counter and range.
    {
    #  The main body of loop - the commands to be repeated.
    }
```

For example, the following will produce the first 21 terms of the arithmetic progression
$a_{t+1} = 100 + 35a_t$

```
1 a<-c()
2 for (n in 0:20)
3    {
4    a<-c(a, 100+n*35)
5    }
```

Here, line 1 creates an empty list `a`. Line 2 declares the loop i.e. it defines a counter `n` that will successively take the values 0 (min) to 20 (max). The brackets in lines 3 and 5 enclose the main body of the loop, the parts that are to be iterated. Several lines of code can be enclosed in the main body but, in this example, we only have one, line 4, which uses the concatenation command `c()`, to add to the current list `a`, the new value `100+n*35`. Since this value contains the counter `n`, it will be calculated anew with each iteration of the loop. To see how the loop operates, try inserting the command `print(a)` as a new line in-between lines 4 and 5. Also, note that the brackets and main body of the loop is typeset towards the right. This makes it easier to identify on the screen or printed page where the loop begins and where it ends.

For this kind of task, where each iteration can be calculated independently of the previous one, a vectorised structure (see Practical I) is preferable to a loop, not only because it is shorter (one line of code instead of 5) but it is also considerably (hundreds of times!) faster. However, when iterations must be done recursively, then a loop is unavoidable. For example, if we were asked to predict the size of a population with logistic growth and no immigration in a constant environment we could use eq.(5). To do this we need some numerical values for the intrinsic growth rate and carrying capacity. We also need a value for the number of founding individuals. Let us say that $r = 0.3$, $K = 100$ and $P_0 = 8$. Here is some code that will generate the first 20 years of the population time series and plot it.

```
1 p<-c(8)
2 for (t in 1:20)
3   {
4   p[t+1]<-p[t]+0.3*(1-p[t]/100)*p[t]
5   }
6 plot(p)
```

An inelegant and potentially time-consuming feature of this code is that the list of population values `p`, is made to grow with each iteration (its first element is specified as 8 and further elements are created in line 4). Ideally, a list of the right length should be created first, and its values simply updated with each iteration of the loop.

■ *Setting out initial conditions and parameters:* It is generally good practice to represent parameters by symbols whose numerical assignments are collected together at the beginning of the code, separately from its main body. This is for three reasons: 1) it gives an overview of the kind of numerical information required for the model, 2) the statement of the model in the main body of the code is more reminiscent of its mathematical description which makes it easier to find mistakes and 3) in many models the same parameter is repeated several times. Rather than having to trawl through the entire code and change the parameter's numerical value in all these instances, it is only necessary to change its numerical assignment at the beginning. Here is a tidier version of the logistic growth model.

```
1  P0<-8        # Initial condition
2  MAX.Y<-11    # Number of years for which to predict
3  R<-0.3       # Intrinsic growth rate (a parameter)
4  K<-100       # Carrying capacity (a parameter)
5  p<-rep(P0, MAX.Y) # List containing the number P0 a total of MAX.Y times
6  for (t in 1:(MAX.Y-1))
7     {
8     p[t+1]<-p[t]+R*(1-p[t]/K)*p[t]
9     }
9  plot(p)
```

4

Since the parameters and initial values are all constants, I have used upper-case letters. Also, here, rather than growing the list in the loop, I create a list of the appropriate length (MAX.Y) and update its elements as part of the loop.

5. Practical tasks

We will implement two models and parameterise them using the life history of red grouse. Model 1 will ignore any constraints to population growth. Model 2 will consider population limitations imposed by habitat availability.

■1■ Read carefully the material in sections 1-4 above.
■2■ Read through the tasks in this section.
■3■ Read through the assessment components in the next section before you carry out any tasks
■4■ Calculate the mortality (d) and recruitment (b) parameters for red grouse using the facts from section 1.

Model 1: Unconstrained population growth
■5■ Assume that there are no constraints to population growth. Write down a model for the growth of this population in discrete time.
■6■ In year zero, a gamekeeper introduces 3 pairs of grouse to an empty moor.
■7■ Calculate analytically, how many birds there will be in the moor after 10 years.
■8■ Write a piece of code to generate the same result by iteration.
■9■ The gamekeeper considers allowing some shooting (annually, a total of 30% of each year's population). Amend your code to incorporate shooting and generate two time series from years 0 to 10: one with and one without shooting.
■10■ Generate a combined plot of these two time series.
■11■ What is the predicted effect of shooting on population dynamics? Could you have obtained this result analytically, if so, how?

Model 2: Density dependent population growth
■12■ For this part, assume that the moor can accommodate a maximum of 200 birds. Write down a model for the growth of this population in discrete time.
■13■ Write a piece of code that will enable you to simulate the population's dynamics from year 1 to 10.
■14■ Plot the resulting time series.
■15■ Now, assume that the gamekeeper decides to actively exclude predation and other sources of mortality, in order to increase the productivity of the moor. The resulting intrinsic growth rate for grouse will depend on the effort he puts into such measures. Plot five time series, each 50 years long, with the following values of r: 1.2, 1.6, 2.2, 2.6, 2.9. Describe what you observe.
■16■ Create two overlapping plots for two populations, both with $r = 2.9$, that start from different initial conditions: in one there are three pairs and in the other four. Describe what you observe.
■17■ The interesting observations in the two previous tasks have been well-documented in the literature as part of a wider dynamic phenomenon. Use your literature search skills to find out what it is called.

6. Assessment

Write a report, no longer than 3 sides of A4, that contains the following:

From model 1:
■1■ Equations of analytical model and its prediction for year 10 (tasks 5 & 7).
■2■ Combined graphs of unrestricted population growth with and without shooting (task 10)
■3■ Answers to the questions in task 11.

From model 2:

■**4**■ Equation for density dependent model (task 12)

■**5**■ Time series plot for moor (task 14)

■**6**■ Time series plots for increasing growth rates and associated comments (task 15)

■**7**■ Combined time series plot for high growth rate and two initial conditions (task 16)

■**8**■ Answer to the question in task 17

■**9**■ Attach your fully commented code from all parts (just the code, no output)