

BB512 - Population Biology and Evolution

Owen R. Jones (Course Coordinator)

2020-09-10

Contents

1	Welcome to BB512	5
1.1	This website and other course materials	5
1.2	Expectations	6
1.3	Your feedback	6
1.4	Assessment	6
1.5	Instructors	6
2	Schedule	9
I	Population Biology	11
3	The legend of Ambalapuzha	13
3.1	Animals/plants, not grains of rice.	14
3.2	Optional: Try these calculations in R	14
4	Geometric growth	17
5	Stochastic population growth	19
5.1	Simulations in R	20
6	Basic logistic population growth	23
6.1	Graph 1	24
6.2	Graphs 2 & 3	24
7	Deeper into logistic growth	27
7.1	Different views of the basic logistic growth model	27
7.2	Adding a time lag	28
7.3	Optional: Cobweb diagrams	29

8	Life tables and survivorship types	31
9	Matrix population modelling	33
9.1	Your turn...	36
9.2	An evolutionary experiment	37
10	Lotka-Volterra competition	39
11	Lotka-Volterra predator-prey dynamics	41
II	Evolution	43
12	Bug hunt camouflage (Netlogo)	45
13	Neutral or adaptive evolution in humans?	47
14	Hardy-Weinberg equilibrium	49
15	How many eggs should a bird lay?	51
16	Trade-offs and the force of selection	53
17	Making a phylogeny	55

Chapter 1

Welcome to BB512

Welcome to the Population and Evolution course. The course is divided into two parts: Population and Evolution.

The population part comes first and covers ecological population dynamics including models of population growth, species interactions, and other demographic models.

The recommended textbook is: Gotelli, NJ (2008) *A Primer of Ecology*. Fourth Edition, Sinauer Associates. ISBN: 978-0878933181

The evolution part comes second and covers microevolutionary processes (natural selection, neutral evolution etc.), population and quantitative genetics, and macroevolution (speciation, extinction and coevolution).

The recommended textbook is: Stearns, SC & Hoekstra, RF (2005) *Evolution: An Introduction*. 2nd Edition, Oxford University Press. ISBN: 978-0199255634

1.1 This website and other course materials

This website is designed to hold most of the materials you need for the practical (mainly computer) exercises you will do during the course. You will also find the Schedule [here](#).

You will find other materials in the folders on Blackboard. These are arranged by week and will include the lecture presentations, handouts etc. Materials will be added as we go along. Apart from the book, there will be some recommended scientific papers to read – these will be accessible via the Dropbox link on Blackboard.

In some of the classes there will be exercises conducted on your personal laptops. Please bring them to class, fully charged.

1.2 Expectations

There are lectures and exercise sessions on the course. The exercise sessions are designed to help you understand the subject better and I expect students to attend and actively participate in both. There will also be some electronic quizzes throughout the semester. These are intended to help you figure out whether you know the material, and whether there are areas you need to revisit. They do not contribute to your final grade, but I hope you will attempt them. They will definitely increase your understanding of the material! Note that the final assessment will be a similar format! I also expect students to make every effort to keep up with the core reading (mainly the textbook chapters), and to ask questions where they don't understand.

1.3 Your feedback

I would really like your feedback on how the course is progressing so I can address any issues that come up as soon as possible. To help with this I have created a simple Google Form: <http://goo.gl/gy2Q6B>. You can use this to send me (Owen) comments (anonymously if you wish) at any time in the course. I promise to do my best to resolve any problems.

1.4 Assessment

The assessment for the course will be an electronic exam held next January with multiple choice and short answer questions. It is worth noting that exam format will be similar to the quizzes mentioned above. The exact date is not yet set.

1.5 Instructors

The instructors of the course are:

- Owen Jones, Associate Professor, jones@biology.sdu.dk
- Thomas Bjørneboe Berg, Associate Prof./Senior Scientist at Naturama, thomas@naturama.dk
- John Jackson, Postdoc, johnja@biology.sdu.dk

Finally, if you have any problems accessing materials, or have any questions regarding the course feel free to send me an email, or make a comment in the form I mentioned above. You can also make an appointment to see me via Zoom or in my office if necessary*.

Owen Jones, course coordinator - jones@biology.sdu.dk

Office location: V12-410b-2

Note: The website is an experimental “work in progress” and will change during the course as I add material. The latest version can always be found at the website.
Please let me know (jones@biology.sdu.dk) if you spot any errors, or have any suggestions for improvement.

Chapter 2

Schedule

This is the schedule for the course. Please note that it is liable to change (possibly at short notice). If you find a mismatch between this schedule and the official one¹, then it is the official one that is correct.

The *Textbook* column tells you the chapter(s) of the two textbooks that are most relevant to the session. The two textbooks are Gotelli (G)² and Stearns and Hoekstra (SH)³, and the numbers refer to the chapter numbers (e.g. SH.3, means Chapter 3 in Stearns and Hoekstra).

You should aim to read these chapters as the course proceeds.

The schedule is only available on the HTML version of this document

```
## tibble [48 x 6] (S3: ical/tbl_df/tbl/data.frame)
## $ DTSTART2 : POSIXct[1:48], format: "2020-09-07 12:00:00" "2020-09-07 13:00:00" "2020-09-1
## $ DTEND2    : POSIXct[1:48], format: "2020-09-07 13:00:00" "2020-09-07 14:00:00" "2020-09-1
## $ Room      : chr [1:48] "U55" "U55" "U23" "U23" ...
## $ Type      : chr [1:48] "Lecture" "Practical" "Lecture" "Practical" ...
## $ Topic     : chr [1:48] "Introduction" "Q&A" "Exponential growth 1" "Legend of Ambalapuzha
## $ Instructor: chr [1:48] "OJ" "OJ" "TBB" "OJ" ...
## - attr(*, "ical")= Named chr [1:29] "VCALENDAR" "GREGORIAN" "PUBLISH" "-//github.com/rianjs
## .. attr(*, "names")= chr [1:29] "BEGIN" "CALSCALE" "METHOD" "PRODID" ...
```

¹<https://mitsdu.sdu.dk/skema/activity/N100007101/e20>

²Gotelli, NJ (2008) *A Primer of Ecology*. Fourth Edition, Sinauer Associates. ISBN: 978-0878933181

³Stearns, SC & Hoekstra, RF (2005) *Evolution: An Introduction*. 2nd Edition, Oxford University Press. ISBN: 978-0199255634

Part I

Population Biology

Chapter 3

The legend of Ambalapuzha

Exponential growth is a powerful concept. To help us grasp it better let's use an ancient Indian chess legend as an example.



According to legend, Lord Krishna once appeared in the form of a wise man in the court of the king and challenged him to a game of chess. The king was a chess enthusiast and naturally accepted the invitation.

The king asked the wise man to choose a prize in case he won. The old man told the king that he had few material needs and that all he wished was a few grains of rice.

He added that the amount of rice itself should be determined using the chessboard in the following manner: one grain of rice would be placed in the first square, two grains in the second square, four in the third square, and so on. Every square would have double the number of grains of its predecessor.

Upon hearing this, the king was unhappy, since the man requested only a few grains of rice instead of any of the other riches of the kingdom, which the king would have been happy to donate (he was a generous guy). He requested the old man to add other items to his prize, but the man declined.

So the game of chess started and, needless to say, the king lost the game so it was soon time to pay the old man his prize. As he started adding grains of rice to the chessboard, the king soon realised the true nature of the wise man's demand. The royal granary soon ran out of rice and the king realised that even

if he provided all the rice in his kingdom and even the whole of India, he would never be able to fulfill the promised reward. He was distraught that he could not fulfill his promise!

Seeing the king upset, the wise man appeared to the king in his true form, that of Lord Krishna. He told the king that he did not have to pay the debt immediately but could pay him over time. The king would serve rice in the temple freely to the pilgrims every day until the debt was paid off. And that is why the Ambalappuzha Temple in India still serves rice to pilgrims – the debt is still being paid off.

Use the Excel sheet (`RiceOnAChessboard.xlsx`) to calculate the quantity of rice that the king owed.
A grain of rice weighs 25mg, what weight of rice did the king owe in total, in kg?

3.1 Animals/plants, not grains of rice.

Imagine that instead of rice, we were talking about the population growth of bacteria, or rabbits, reproducing every time step.

- Would the model be realistic?
- Why/why not?
- What other factors should be taken into account?

3.2 Optional: Try these calculations in R

You can do this kind of calculation easily in *R*. Try this.

```
myData <- data.frame(Squares = 1:64, nRice = NA)
myData$nRice[1] <- 1

for (i in 2:64){
  myData$nRice[i] <- myData$nRice[i-1]*2
}
```

Now we can look at the top and bottom of the 64 row data frame like this:

```
head(myData)
```

```
##   Squares nRice
## 1         1     1
```

```
## 2      2      2
## 3      3      4
## 4      4      8
## 5      5     16
## 6      6     32
```

```
tail(myData)
```

```
##      Squares      nRice
## 59      59 2.882304e+17
## 60      60 5.764608e+17
## 61      61 1.152922e+18
## 62      62 2.305843e+18
## 63      63 4.611686e+18
## 64      64 9.223372e+18
```

And we can sum up the total number of grains of rice on the 64 squares of the board like this:

```
sum(myData$nRice)
```

```
## [1] 1.844674e+19
```

To put that *HUGE* number in context, if a grain of rice weighs 25mg (0.0000025kg), then we'd have 46,116,860,184,274 kg.

Chapter 4

Geometric growth

In a simple model of population growth where the population grows without any constraints, the speed a population increases in size can be described by the population growth rate. This is often given by the symbol lambda (λ). High values of λ mean the population grows fast, while small values indicated that it grows slowly, or shrinks in size.

Geometric growth is often seen as synonymous to exponential growth. It basically is, but with one important difference: exponential growth, strictly speaking, refers to “continuous time” scenarios, whereas geometric growth refers to models where the population changes in discrete time steps (e.g. each year).

Download and open the Excel file `GeometricGrowth.xlsx`.

1. Starting with an initial population size (N) of 10 [at time (τ) =0], and with a λ of 1.1, use Excel’s equation functions to work out the population size from $\tau=1$ through to $\tau=20$. E.g. The formula might look something like this “=B8*\$F\$8”.
2. Use charts to plot the results. On the horizontal axis (x-axis) you should have time, and on the vertical axis (y-axis) you should have population size (N). A scatterplot would be best for this.
3. Make another plot of the same data, but this time use a natural logarithmic axis for the population size. The formula in Excel for natural log is LN().
4. What do you notice about the curves in these different versions of plotting the same data?
5. Try altering the population growth rate. Try values of 0.8, 1 and 1.2 for example. What happens to the curves? What is special about the value of 1?

Now reset the population growth rate to 1.1. Now lets see if the mathematical rules of so-called “geometric series” work...

1. If the population starts at $t=0$, $N_t = N_0 \lambda^t$. In words, the population at time t is the initial population size multiplied by the finite population growth rate raised to the power of t .

2. Your instructor will explain how this works on the blackboard.
3. Following these rules, the population size at time $t = 5$ (N_5), is $10 * 1.1^5$, or $10 * 1.61051$, which is 16.1051. Check that this matches what you got earlier.
4. This approach can be really useful because it can save lots of time. For example, what is the population after 900 generations? It would be tedious to work this out using the first, “simple” approach, but very easy if you use the geometric series rules.

Chapter 5

Stochastic population growth

We will examine the effect of adding stochasticity (randomness) into the simple exponential/geometric growth model you have been looking at in the last couple of lectures. Remember – this model allows for unbounded population growth – the populations development is not influenced by population density.

We'll focus on the discrete form: $N_{t+1} = \lambda N_t$

In many cases it is unlikely that lambda (λ) will be constant through time. It is likely to vary because of environmental factors (weather, food supply etc.) and internal factors (structure of the population etc.).

What effect might this variation have?

A good way to begin consider this question is to think of lambda as a random number drawn from a particular distribution (Fig. 5.1). In the figure you can see that in good years λ is > 1 so the population will increase, in poor years λ is < 1 , so the population will decline. In most years, the population will neither grow nor shrink.

When using the equation above to calculate population at time $t + 1$ (N_{t+1}) from the population at time t (N_t), one would draw a random λ value from this distribution. Sometimes λ will be high, other times it will be low, most of the time it will be from around the middle of the distribution.

Use the Excel worksheet, **StochasticPopulationGrowth.xlsx**, to study how stochastic population growth works with this simple model.

- 1) Use Excel formulae to calculate population growth through time (20 generations, with starting population of 100), linking to the mean finite population growth rate.
- 2) Use charts to plot the results. (you already did this last time!)

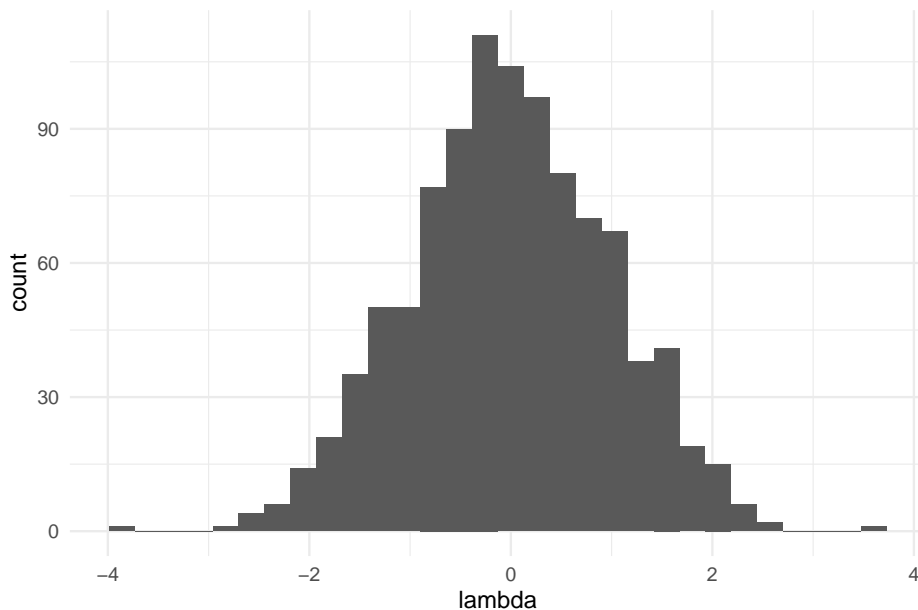


Figure 5.1: A normal distribution of potential lambda values

- 3) Use a formula to generate a column of stochastic lambda values. [English Excel: `=NORMINV(RAND(),F10,SQRT(F11))` / Danish Excel: `=NORMINV(SLUMP();F10;KVR0D(F11))`]
- 4) Use the same procedure as before, to create a stochastic population size vector (stochastic N)
- 5) Compare the two trajectories using a chart.
- 6) Try altering initial population size, the mean finite growth rate, and the amount of stochasticity (Variance).
- 7) Extinction occurs when $N \leq 0$. What happens to extinction risk as stochasticity increases? What about when initial N is small?

5.1 Simulations in R

Excel is of limited use to really get a feel for this. For the next part we'll use *R*.

If you already have R on your computers you can play along, otherwise take a look at my demonstration in class. I will show how you can use this simulation approach to estimate extinction risk and how this is related to starting population size, mean lambda, and the amount of stochasticity.

You can copy/paste the code below into *R*.

The output of the modelling is shown in Fig. 5.2

Copy-and-paste the code below into a text file (or directly into R). The final line of the code (`nExtinct/nTrials`) gives you an estimate of extinction probability - the proportion of trials that lead to a population size of 0 (or less). Modify the simulation settings to explore what happens to (i) the plot of population growth and (ii) extinction risk, when you vary the amount of stochasticity (`var.pgr`), the deterministic λ (`pgr`) and the number of generations (`nGen`).

```
#Simulating stochastic geometric population growth rate

#Simulation settings (try changing these)
pgr = 1.05 # deterministic growth rate
var.pgr = 0.1 # amount of stochasticity
startPop = 10 # pop size at start
nGen = 50 # number of generations
nTrials = 100 # number of repeated simulations

#####
#If you are unfamiliar with R, do not edit anything below this line!
#####

pseudoExtinction = 1

# First randomly generate some lambda values
lambdas<-matrix(rmnorm(nTrials*nGen, mean = pgr, sd = sqrt(var.pgr)),ncol=nTrials,nrow=nGen)

# Use a histogram to see what they look like (uncomment the line below)
# hist(lambdas,col="grey",main="")

# Now run the simulations to see what the resulting population growth looks like
trials = matrix(data = NA, nrow = nGen, ncol = nTrials)
for (j in 1:nTrials){
  popSize = startPop
  for (i in 2: nGen){
    stoch.pgr = lambdas[i,j]
    popSize = append(popSize, popSize[i-1]*stoch.pgr)
  }
  trials[,j] = popSize
  rm(popSize)
}

#Calculate probability of (pseudo)extinction
minvals <- apply(trials,2,min)
nExtinct <- length(minvals[minvals<=pseudoExtinction])
nExtinct/nTrials
```

```
## [1] 0.34
```

```
#Make a plot of the population trajectories
plot(1:nGen,log(seq(0.1,max(trials),length.out=nGen)),type = "n",axes=F,xlab = "Time",
matlines(log(trials),col = "#FF234520",lty=1,lwd=3)
axis(1)
axis(2,at = log(c(0.1,1,10,100,1000,10000,100000)),
      label = c(0.1,1,10,100,1000,10000,100000))
abline(h=log(pseudoExtinction),lty=2)
```

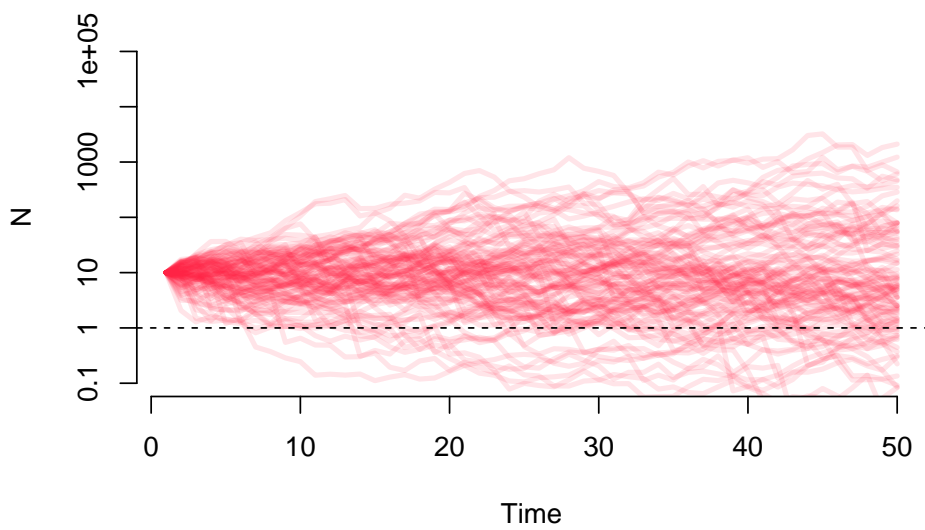


Figure 5.2: An example of stochastic population projection (100 simulations for 50 generations)

Chapter 6

Basic logistic population growth

Population sizes have upper limits – they can only get so large. This is often modeled with the ‘logistic growth’ model¹:

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

This equation models population at time $t + 1$ (N_{t+1}) as a function of the population at time t (N_t), the *discrete growth factor* (r_d), and carrying capacity of the environment (K).

The idea is that the growth rate of the population (the difference between N_{t+1} and N_t) decreases as the population increases in size. When the population size at N_t is equal to the *carrying capacity* of the environment (K), the population growth rate is zero. For example, if the carrying capacity of the population is 200, and there are already 200 individuals in the population, then the size of the population will remain unchanged through time ($N_{t+1} = N_t = 200$).

The aim of this Excel-based exercise is to explore this model and help you get an intuitive understanding of it by looking at it from different perspectives. Even though it is a fairly simple model, it leads us to some useful biological insights.

Open the file **Basic Logistic Growth.xlsx**.

You will see that there are three blocks of numbers, and three graphs. During the exercise you only need to edit the pink block.

The pink block gives the important parameters of the logistic model:

- **Initial N** = the starting population size at time 1.
- **r_d** = the maximum per capita population growth rate (r_d). When r_d is 0, the population does not grow. When r_d is > 0 the population grows, and when it is < 0 it declines. The population cannot fall below 0.
- **K** = the carrying capacity of the population.

¹This is equation 2.4 in Gotelli

The initial values for these are 10 , 0.8 and 200 respectively.

In this exercises you will be altering these parameters and observing the outcome in the 3 graphs which show:

1. The population size through time.
2. The per capita growth rate of the population in relation to population size
3. The relationship between population size at time t and at time $t+1$

6.1 Graph 1

First – take a look at **Graph 1**.

- What is the maximum population size?
- How does this compare to carrying capacity (K)?
- What do you predict to happen if you increase K to 300 ? (try it)

- At what time do you reach the maximum population size?
- If you halved the discrete growth factor (r_d) to 0.4 , what do you think will happen to the dynamics this time? (try it)

- What do you think will happen if you r_d to 1.8 ? (try it)
- What do you notice about the population size through time?
- How does the maximum population size compare with the carrying capacity? How would you describe the ‘*dynamics*’ of this population?

- What happens if you increase the r_d even more, to 2 or 2.1 ?
- And even more to 2.8 , 2.9 or 3 ?
- How would you describe these dynamics?
- Is there a limit to how high r_d can be? (hint: populations go extinct if $N < 0$).

Finally, compare the population trajectory in **Graph 1** for populations with $r_d = 2.8$ and 2.81 . Then compare the trajectories where you fix r_d at 2.8 but vary initial population size by a small amount (e.g. 1). Imagine you were a population manager – would these populations be easy or hard to predict? What kinds of species have high population growth rates like these?

6.2 Graphs 2 & 3

Now let’s turn to **Graphs 2** and **3**.

In **Graph 2**, notice that the per capita growth rate always declines linearly with population size (N). Where does it cross the x-axis line? Modify the carrying capacity (K) – what do you notice?

Graph 3 is shaped like a parabola, starting with small values, increasing to a maximum, then declining to small values again. The position of the maximum is dependent on the values of K and r_d . When r_d is small (around 1), the peak is at $x = K$ and $y = K$. Explore (1) how the peak moves around if you fix one of K or r_d and alter the other parameter and (2) how the slope of the line changes at $N_t = K$.

Hint: the slope of the line changes from positive to negative as r_d increases.

These three graphs are simply different ways to visualize the same model. It is important that you can make the connections between these graphs.

How would the same plots look for regular exponential growth?

Some useful keywords:

- Oscillation
- Cycle/cyclic dynamics
- Stable-limit cycle (2-point, 3-point limit cycle)
- Chaos/Chaotic dynamics
- Unpredictable/predictable

Chapter 7

Deeper into logistic growth

The purpose of this exercise is (i) to look at the relationships between the exponential (or geometric) growth model and the logistic growth model and (ii) to emphasise different ways of looking at the models.

I will begin by showing the equations on the (actual!) blackboard for exponential and logistic growth. I will highlight how the logistic model is a simple extension of the exponential model. I will show that when K is infinite the model $\frac{dN}{dt} = rN \left(1 - \frac{N}{K}\right)$, simplifies to the exponential model ($\frac{dN}{dt} = rN$), because the $\left(1 - \frac{N}{K}\right)$ bit drops out of the equation.¹

The same is true for the discrete version of the logistic growth model: Compare the equation $N_{t+1} = N_t + r_d N_t \left(1 - \frac{N_t}{K}\right)$ with $N_{t+1} = N_t + r_d N_t$.²

You can then look at the Excel file `Deeper Into Logistic Growth.xlsx`

7.1 Different views of the basic logistic growth model

Take a look at the `BasicLogistic` Excel worksheet/ark.

Focus first on **Figure 1**.

Enter different values for `r_d`, e.g. 0.8, 1.2, 1.8, 2.4, 2.7...

Can you describe the dynamics of the population time series in Figure 1 using some of the following vocabulary: Oscillation, damped oscillation, stable cycle, 2-point cycle, chaotic, unpredictable, predictable?

Now focus on **Figure 2**.

This figure shows the per capita population growth rate as a function of population size at time t . Note where the line intercepts the x- and y-axes.

¹These are equations 2.1 and 1.1 in Gotelli.

²These are Expression 1.13 (page. 11) and equation 2.4 from Gotelli.

What do you notice about these values? Hint: what are the values you have set for r_d and K ? Try varying those values.

On paper, make a sketch of a graph for per capita population growth rate as a function of population size at time t for a logistic model with $r_d = 1.5$ and $K = 250$, indicating the values of the intercepts. Check your sketch by entering those values into the Excel model.

Based on what you know about what happens to the population dynamics for different values of r_d and what you have just seen in Figure 2, you should now be able to sketch fairly accurate cartoon graphs if you are given values for r_d and K !

Without using Excel, draw a population time series, and a graph of per capita growth rate vs. N for when initial population size is 10, r_d is 1.9 and K is 500. Check your graph using Excel.

Focus now on **Figure 3**.

Population growth rate is dN/dt - the rate of change in N (dN , change in population size) per unit time (dt). You should now explore how *Figure 3* relates to the values used in the equation by changing the r_d and K values and looking at *Figure 3*.

Can you see how *Figures 1, 2 and 3* are connected?

At what population sizes is the population growth rate 0 (dN/dt)? What is the population size at which the population growth is rate maximized?

Now lets look back on **Exponential growth**.

How do the relationships in Figures 1, 2 and 3, differ from the equivalent figures for logistic growth. Try to draw graphs of Figures 1, 2 and 3 for the exponential growth model.

- Fig 1. Population size (N) through time (t).
- Fig 2. Per capita growth rate ($\frac{1}{N} \frac{dN}{dt}$) vs. population size (N)
- Fig 3. Population growth rate ($\frac{dN}{dt}$) vs. population size (N) (see figure 2.4 in Gotelli)

7.2 Adding a time lag

This relates to *Equation 2.3* in Gotelli (and pages 32-35).

$$\frac{dN}{dt} = rN \left(1 - \frac{N_{t-\tau}}{K} \right)$$

Look at the Excel Worksheet/Ark called **TimeLag**.

Adding a time lag to logistic model can complicate the dynamics (by introducing cycling). The 3rd Excel tab has an exercise showing that if you add in a time lag to the logistic function it modifies the dynamics. The purpose of this Excel sheet is to allow you to prove this to yourself!

You will need to carefully modify the Excel formula so that instead of referring back to the population size at N_{t-1} , it refers back to N_{t-1} etc. Remember to

drag the formula down to all the other years (or use the shortcut I will show you).

Start with a small r_d that gives a smooth convergence to K with an ordinary logistic model. Add in a 1-year time lag and show that this generates cyclic dynamics. This shows that this simple “quirk of life history” (a time lag) can generate cycling, even if the population growth rate is low.

7.3 Optional: Cobweb diagrams

The cobweb diagram is another useful tool to visualize and explore dynamics of logistic models. See the book section by Mathiopoulos in Blackboard. They would be very hard to implement in Excel, so I have made a webapp: (<https://jonesor.shinyapps.io/cobweb/>)

The graph shows a “track” which follows the fate of a population. The track bounces between the parabola describing the relationship between N_t and N_{t+1} (Figure 3.1 in the excel sheet) and a 1:1 line. Try altering r_d in the model and observe what different types of dynamics look like with this “view”. Check out what happens if you modify the starting population size. You should see that for non-chaotic dynamics, the starting population size doesn’t affect the fate of the population. For example, set initial population to be 0.01, and r_d to be 0.9. You should see damped oscillations. Now move the initial population slider. You should see that the population always ends up with the same dynamics, heading towards carrying capacity.

Chapter 8

Life tables and survivorship types

Life tables are tables that shows for each age, the probability that an individual of that age will die before the next birthday (*probability of death*). This exercise deals with so-called **cohort life tables** which, as the name implies, follows a “*cohort*” of individuals from birth until they all die. A cohort is the group of individuals born within a particular time interval (e.g. “*all individuals born in 1998*”).

Life tables have been used extensively in population biology, in human demography and in epidemiology. They are also important outside of biology, e.g. in the management of product life-cycles, such as in cars or other machinery.

The basic algebra used in life tables is explained in Gotelli Chapter 3 (see table 3.1 for an example).

The purpose of this exercise is:

1. to allow you to calculate a life table yourself in Excel;
2. to develop your Excel modeling skills by asking you to make the calculations following the equations given in Gotelli (and in the lecture slides);
3. to allow you to explore different types of survivorship (Type I, II and III) and consider the relationship between these life tables and “life history strategy” (more on this in the evolution part of the course).

Open the Excel file **Life tables exercises.xlsx**.

The file has three worksheets (“*Life table*”, “*Survivorship Curves*” and “*Gotelli Table 3.1 example*”).

Let’s start with “**Life table**”.

The aim now is to use Excel as a modeling tool to produce a life table. I have provided some initial data collected from a cohort of animals. I know how many individuals survive each year (how many “enter the interval”). I also know how many babies (on average) are produced by each female.

Start by calculating survivorship (l_x). Survivorship is the **probability of survival to a particular age**. Therefore, at time 0, $l_0 = 1$, since everyone is alive at this point. The next value (l_1) must be calculated based on the number alive at that point. In this case it is $352/500 = 0.704$. You must generalize this calculation into a formula that can be dragged to fill column D in the worksheet. In algebraic form, the equation is $l_x = S_x/S_0$.

Next, you can calculate age specific **survival probability**. Note that this is different from l_x . Survival probability is simply the probability that an individual will survive its current age class. i.e. what is the probability that an individual currently aged 2 will survive to become age 3. In this case, the $254/298 = 0.852$. The calculation is $g_x = l_{x+1}/l_x$, or S_{x+1}/S_x .

Now complete the remaining two columns, and use them to calculate (a) R_0 ; (b) **Generation time**; and an approximation of r . **Tip** You need to understand the use of the \$ symbol in Excel, and how to drag the selected area to place the formula in the column. Refer to the sheet “*Gotelli Table 3.1 example*” if you get stuck (you should be able to see the formulae used there).

In the second part we focus on the **Survivorship Curves** worksheet. The aim here is to start to explore how different types of organisms with different ways of life (“**life history strategies**”) can have qualitatively different kinds of life tables. The most important thing to observe is the difference in **survivorship curves** (l_x). These changes become very obvious when you plot the log-transformed survivorship against age.

In the Excel worksheet, I have placed tables showing the fate of cohorts of three populations of different species. Your job now is to calculate the survivorship curve (l_x) for these species, take the natural log (using formula `=log(C3)`, for the first population, `=log(H3)` for the second population etc.

You should see that the graphs automatically fill up with lines. These show Type I, II and III survivorship.

Chapter 9

Matrix population modelling

Think of an organism you would like to model the dynamics of. It could be a mammal, a bird, a fish, insect or tree ... real or fantasy.

Think about their life cycle, and draw it as a life cycle diagram with circles indicating the stages and arrows representing transitions between stages and reproduction. Next to the arrows, write values for survival probability and fecundity (number of babies) using your biological knowledge.

Things to think about:

- Is it age based or stage based?
- How many stages are there?
- If it is stage, how are stages defined? E.g. by size, by development, etc.
- Are the survival and fecundity higher in earlier or later life?
- Does the organism skip stages?
- Does the organism move backwards through the life cycle?

Turn this diagram into a matrix population model by filling in a square of survival/fecundity values. There is space below for up to a 4-stage matrix model.

Working with matrices is very tedious in Excel. However, in R you can use this information to predict the future dynamics of the population, and estimate population growth rate, and generation time etc.

Open up **RStudio**, and lets see if we can predict future dynamics. First you will need to install a package called **popdemo**.

```
install.packages("popdemo")
```

You only need to install packages once. After that you can load the package for use by using the **library** function.

```
library(popdemo)
```

You can put your matrix into R like in the example below (change the numbers to match YOUR model). If your model has fewer, or more, stages then you will need to modify the code a bit. Ask for help if you get stuck.

```
A <- matrix(c( 0.00, 0.00, 4.00, 2.00,
               0.10, 0.00, 0.00, 0.00,
               0.50, 0.20, 0.00, 0.00,
               0.00, 0.30, 0.40, 0.30),
            byrow = TRUE, nrow = 4)
```

And now you can use the **project** function to project what happens to the population, then plot it. Look at what happens if you log or don't log the y-axis. First you need to define an initial starting population structure.

In my example, I have 4 stages, so I have 4 values for the initial population sizes. Then I use the `popdemo` function `project` to do a population projection for 10 time steps.

```
initial <- c(10,5,3,3)
pr <- popdemo::project(A, vector = initial, time=10)
```

Take a look at `pr`, the projected population. This gives you the total population size, and below that the population sizes in each stage.

```
pr
```

```
## 1 deterministic population projection over 10 time intervals.
##
## [1] 21.0000 28.6000 45.9800 68.7940 110.7142 167.8147 267.2773 408.6193 646.18
```

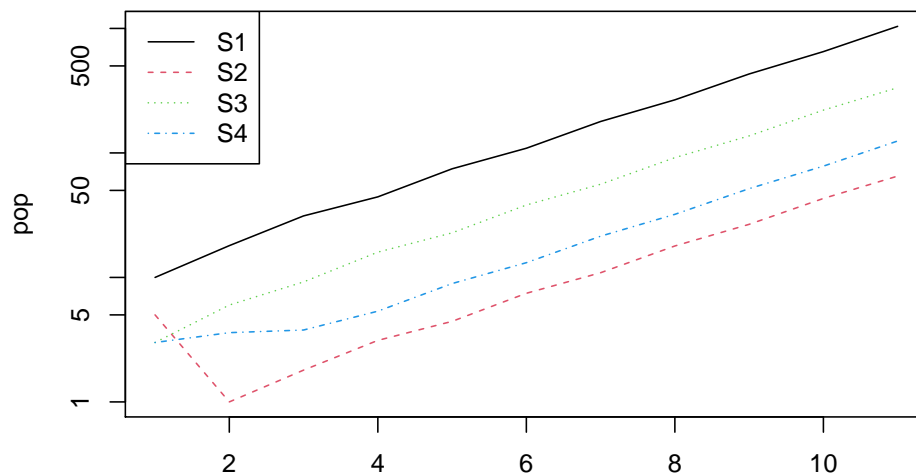
You can access the population sizes of the different stages using `vec(pr)`.

```
vec(pr)
```

```
##           S1           S2           S3           S4
## [1,] 10.0000  5.00000  3.00000  3.00000
## [2,] 18.0000  1.00000  6.00000  3.60000
## [3,] 31.2000  1.80000  9.20000  3.78000
## [4,] 44.3600  3.12000 15.96000  5.35400
## [5,] 74.5480  4.43600 22.80400  8.92620
## [6,] 109.0684  7.45480 38.16120 13.13026
## [7,] 178.9053 10.90684 56.02516 21.44000
## [8,] 266.9806 17.89053 91.63403 32.11412
## [9,] 430.7643 26.69806 137.06842 51.65501
## [10,] 651.5837 43.07643 220.72178 78.33329
## [11,] 1039.5537 65.15837 334.40714 124.71163
```

Let's plot this...

```
pop <- vec(pr)
matplot(pop,type="l",log="y")
legend("topleft",legend = colnames(pop),col=1:ncol(pop),lty=1:ncol(pop))
```



You should see that the population increases exponentially. The population growth rate is the so-called “*dominant eigenvalue*” of the matrix \mathbf{A} .

We can ask R for the *eigen values* and *eigen vectors*. These are the population growth rate (λ) and the stable stage distribution (*SSD*) and the reproductive values (*RV*) of the different stages. *SSD* is the expected *proportion* of individuals in the different stage classes at equilibrium and *RV* is the expected number of future offspring by individuals in each stage.

You can see that in this case, using my example values the population is growing 55.74% per year.

```
eigs(A)
```

```
## $lambda
## [1] 1.557365
##
## $ss
## [1] 0.66068003 0.04242295 0.21756281 0.07933420
##
## $rv
## [1] 0.6831674 0.4705100 2.0337794 1.0866656
```

9.1 Your turn...

- 1) by editing the inputs in the code above, make a projection for **your** data.
- 2) plot the results (using the R code here)
- 3) what is the health of your population?

9.2 An evolutionary experiment

You can think of λ (population growth rate) as being a measure of fitness. Imagine that some of your population had a mutation that caused them to have, say, 1 extra baby, but at the expense of reduced survival in one of the younger stages. Would this mutation persist in the population?

Chapter 10

Lotka-Volterra competition

[coming soon]

Chapter 11

Lotka-Volterra predator-prey dynamics

[coming soon]

Part II

Evolution

Chapter 12

Bug hunt camouflage (Netlogo)

[coming soon]

Chapter 13

Neutral or adaptive evolution in humans?

[coming soon]

Chapter 14

Hardy-Weinberg equilibrium

[coming soon]

Chapter 15

How many eggs should a bird lay?

[coming soon]

Chapter 16

Trade-offs and the force of selection

[coming soon]

Chapter 17

Making a phylogeny

[coming soon]