**This lab is about modelling ecological population dynamics, which means the way that populations change over time. You are encouraged to collaborate with others in working through this lab, but do make sure that you fully understand what is going on.**

**This lab will be closely related to a questions on the assignment. You will answer the questions on the assignment (about this lab) through entering answers into an online LMS ‘quiz’. You will not be able to access the quiz itself during the labs – it will open later. As you go through this lab sheet, write down notes in the space provided for the questions marked with a \*, either in pen or in the word document. These will help you check your understanding as you go and form the basis of your answers for the assignment quiz. Pasting plots into this word doc can also be very useful, so you can look back at them to help with the quiz later. When you start writing your R script, make sure it is neat and structured, as you will have to submit that too. But you will not have to submit this document or any written report on this lab.**

**Part 1: Population Dynamics and Competition**

While the ideas of modelling population dynamics will be very similar for many different species and situations, for this exercise we can assume that we are modelling the density of a species of mouse, that the time step is in months, and the density is the number of individuals per square kilometre. For other species we might use different spatial and temporal scales.

First we will work with the ‘intro ecological modelling.R’ file from LMS. There are some basic intro/revision things at the start. Work through those and check that you understand the loops and the creation of the vectors. Ask questions about anything that isn’t clear. Work through until you’ve got to the unbounded growth model, and check you understand how that works.

Look at the plot of time against density, so you can see how density is changing over time.

\*What do you notice?

Population density is increasing over time. Seems like a steady increase rather than exponential growth. Positive relationship.

A graph with purple dots

Description automatically generated

Look at the formula

pop[time] = pop[time-1] + growthrate \* pop[time-1].

This means that the new population density is the old one plus a certain proportion of the old one. This proportion is the reproduction rate; a value of 0.1 means that, on average, each mouse in the population produces 0.1 babies per month. Try changing this rate and see what happens in the graph (You’ll need to rerun the model and replot the results each time you change something of course).

\*What happens when you increase the reproduction rate?

By changing population rate to 0.3 there is a stronger curve developing. Over time the population density is increasing at an accelerated rate

A graph with a line drawn on it

Description automatically generated

In more mathematical notation this equation could be written as:

dt+1= dt + r dt

Now extend the model to time=100. Also extend the graph to time=100. What do you see now?

Huge exponential growth particularly towards the end of the century

A graph with a line

Description automatically generated

\* What happens if you reduce the reproduction rate? (Change it back to 0.1 when you’re done.)

Curve Flattens out (reproduction rate 0.05)

A blue line graph with numbers

Description automatically generated

\*Why is this not a realistic model of population growth?

*Doesn’t consider death rates/natural mortality. Also emigration (away)*

One thing we haven’t accounted for is the fact that animals die as well as reproduce. Add a death rate parameter equal to 0.05 to the R code. Adapt the equation to account for this death rate as well as the reproduction (birth) rate.

\*What happens?

*When births are equal to deaths the population density will stay constant*

At the moment we are assuming that death rate is constant, but in reality it is likely to increase as the population gets large and resources become limiting. Instead of a constant death rate, we now want a death rate that changes with population size. Assume that death rate will be equal to 0.0001 times the population density. Let’s call the parameter 0.0001 the ‘density dependence death rate parameter’. (Introducing a density-dependent death rate creates a ‘bounded growth model’. See the hints in the R script for creating this model as well. And if you get stuck there are a couple of solutions as well.)

\*What happens?

*The population boom will level off as the carrying capacity is reached due to resource limitation amongst other things I’m sure*

A graph with a red line

Description automatically generated

\*Why is this model of population growth more realistic?

*Resources aren’t infinite. Once resources become stretched and limited there will be emigration, increased mortality and reduced fecundity.*

Note that the population tends towards a certain density that we can call the equilibrium population density.

# this means at what point is it levelling out and reaching the carrying capacity.

\*About how long does it take for the population to get close to its equilibrium population density?

*Time = 90 (assuming years)*

\*What happens if the initial population density is higher than the equilibrium population density?

*Mass die off until equilibrium reachedA graph with a red line

Description automatically generated*

\*What happens if the initial population density is equal to the equilibrium population density? (Then set the initial population density back to what it was to start with.)

*Population starts and maintains equilibrium*

A graph with a red line

Description automatically generated\*What happens now if you increase the ‘reproduction rate’ parameter?

*The equilibrium is at a higher density (~ 5000) and it is reached in just over 20 years*

A graph with red line

Description automatically generated

\*What happens now if you decrease the ‘reproduction rate’ parameter?

*Much slower increase in population density. Equilibrium not reached in the first century*

*A graph with a red line

Description automatically generated*

\*What happens if you change the ‘death rate’ to be 0.0002 times the population density instead of 0.0001 (ie change the ‘density dependence death rate parameter’ to be equal to 0.0002)?

*Equilibrium reached in approximately the same amount of time as previous (where growth rate = 0.1 and population density began at 5) which is about 85/90 years. However, carrying capacity aka the density equilibrium is halved to 500*

A graph with a red line

Description automatically generated

\*How does the equilibrium population density depend on the ‘reproduction rate’ parameter and the ‘density dependence death rate parameter’?? Can you write down an equation or give an explanation in words that actually gives the equilibrium population density as a function of the ‘reproduction rate’ parameter and the ‘density dependence death rate parameter’??

*When the population is at equilibrium the birth rate and death rate are equal.*

*could write some algebra here to maths it out*

RESOURCE LIMITING MODEL

Now we want to do something similar but taking resource levels and consumption into account more explicitly, so we can start to think about competition. (For the mouse – the main limiting resource is food, edible seeds produced by plants in its environment).

We’ll create a new model, representing a situation where a small population of the mouse species colonises a new environment (an island perhaps). It is probably easiest to start by copying and pasting the code you wrote already (your model) and add a comment for yourself so you know this is the mouse competition model when you look back later. We are changing the time step for this new model – the time step is now days.

1. Add a new variable called ‘food’ – this is the biomass of edible seeds available (kg per square kilometre). Set the initial value for this variable to be 100.
2. This mouse needs at least 10g of food per day to just stay healthy. Within the loop, we can represent this by adding the following line of R code at the start of the { } block:
   1. foodneeded = pop[time-1] \* 0.01
3. The amount of food eaten on any day will depend on how much food the mouse finds – more mice will find more food, and if there is more food then more will be found. We can represent this with the following line of code:
   1. foodeaten = food[time-1] \* pop[time-1] \* 0.001

\* If the population was twice as big on a given day, and the amount of food was the same, what would happen to the amount of food eaten, according to this equation (note that this question is just checking to see if you understand the equation above – you don’t actually need to run or plot anything) ?

*the amount of food eaten on a given day is equal to the amount of food left over from the day before, multiplied by the population size of the day before \* 0.001. Therefore twice as many mice would eat twice as much food.*

\* If the population was twice as big, and the amount of food was also twice as big, what would happen to the amount of food eaten, according to this equation?

*Quadruples the amount of food eaten*

The amount of food left at the end of a day will be equal to the amount of food left at the end of the previous day, minus the amount of food eaten on this day. Add a line of code to represent this.

*food[time] = food[time-1] – foodeaten*

*where foodeaten is as above.*

The growth rate of the population will be positive if the mouse finds more than enough food, zero if it finds just enough, and negative if it doesn’t find enough. Add the following line of code to represent this:

growthrate = 0.05 \* (foodeaten - foodneeded)

*I’m assuming that the 0.05 is representing reproduction where enough resources are available?*

The population at the end of a day should be the population at the end of the previous day, plus the change in population, and the change in population should be the population at the end of the previous day times the growth rate. Make sure that’s what your code is saying (there is no need for a death rate now, since the growth rate can be negative).

pop[time] = pop[time-1] + growthrate \* pop[time-1]

Run your new model over 365 days. Plot time against population, like before. Make sure the y-axis includes zero – you can do this by adding the option ylim=c(0,150) to the plot command. Add the food to the plot in red with the following command:

lines(1:maxtime , food,col='red',lwd=2,lty=2)

\* What is happening on this island after the first mice arrive. What happens to the mice and what happens to the food?

*in the first 100 days the mice population experiences is a significant boom and reaches a maximum density of 133 individuals. However, available food on the island quickly depletes leading to a population crash shortly before 100 days. Eventually the population reaches equilibrium of ~ 5 individuals nearing the end of the year*

*A graph of food and population

Description automatically generated*

\* For mice to survive and reproduce, they need to find mates and companions. This means there is an ‘Allee Effect’ – at small population densities, populations are no longer viable. We can represent this in a simple way by adding the following line of code at the end of the block:

if (pop[time]<3) pop[time]=0

\* What is the minimum viable population density for this mouse?

*3 individuals assuming from the code*

\* How long does it take for mice to die out completely? (You might need to extend the simulation.)

*749 days*

\* If each mouse needs a little more food each day, does that make them die out earlier or later?

*Earlier*

\* If each mouse is a little better at finding food each day, does that make them die out earlier or later?

*Earlier – as they’ll find food resources sooner and exploit them earlier, leading to a boom in population putting greater pressure on available food resources*

\* If the growth rate is a little smaller, does that make them die out earlier or later?

*Later – after 1000 days*

\* If the amount of food initially available is a little less than 100kg, does that make them die out earlier or later?

*Earlier – but only slightly earlier. Has a stronger effect on population growth rate*

\* What makes more difference to how long they last – halving the amount of food initially available, or doubling the initial population?

*Half Food = 771 days til Unviable population*

***Double Population = 697 days***

\* Make sure you set all your parameter values back after changing them – have you done that?

Of course the grass on the island actually keeps on growing and producing seed. Each day the grass produces half a kilogram of edible seed. Make this change to the model.

*food[time] = food[time-1] - foodeaten + 0.5*

\* What is happening on this island after the first mice arrive now? What happens to the mice and what happens to the food?

A graph of a number of extinguishing a number of days

Description automatically generated with medium confidence A graph of a seed

Description automatically generated

*Greater food availability => boom in growth rate to 2.5x greater population size then significant bust followed by equilibrium of food reached at about 220 days with pop density of 49 sustained on the available food (10kg/km2)*

# Second Invader

Now we want to model what happens if a second species of mouse colonises the island a year after the first one arrives. To do this we need a new population variable (call it ‘pop2’). The initial population of the second species is zero, but at the end of day 365, some mice arrive. Let’s say 60 arrive, and the area of grassland on our island is 10 square kilometres, so on day 365 the population density of the new species jumps to 6. Include this in your model using an ‘if’ command.

*pop2 = 6 at time[365]*

At this point, run the model, and add another line to the output plot showing the density of the second species over time. You see it jump to 6 on day 365, then remain constant.

A graph of a number of invaders

Description automatically generated

This second species eats the same seeds as the first one. It needs the same amount of seeds per day to just remain healthy. It is exactly as good at finding food, has the same growth rate, and this growth rate depends on the food it finds and eats in the same way.

Adapt the model as required – you will need to copy and paste and modify most of the lines of code – be careful that you change all relevant variable names in each line.

When you’ve done that, run the model and check it is working correctly.

\* What happens to the population of the new mouse after colonisation?

*Population declines to ~5.91per/km2 then remains stable*

*A graph of a number of invaders

Description automatically generated*

\* What happens to the population of the first mouse after colonisation by the new mouse?

*pop density equilibrium drops to 44 from 50 and stays relatively stable at 44.01 mice per km2*

\* What happens to the amount of food available after colonisation by the new mouse?

*Drops slightly by 0.5kg/km2 before returning to slightly higher than previous equilibrium at 10.01kg/km2.*

Now change the model so that the new mouse is bigger than the old one and thus needs more feed each day ie 15g instead of 10g.

When you’ve done that, run the model and check it is working correctly.

A graph of food requirements

Description automatically generated

\* What happens to the population of the new mouse after colonisation?

*Population declines from initial colonisation by just under half at time = 1000.*

\* What happens to the population of the first mouse after colonisation by the new mouse?

*Small decline in population density, reaching lowest point at around 120 days following colonisation (time = 585, pop density = 44.6). Population density then slowly increases over time, reaching 46.8 at time = 1000.*

\* What happens to the amount of food available after colonisation by the new mouse?

*Declines initially from 9.995 to 9.27 at 32 days following colonisation before increasing over time to 10.08 approx. kg/km2 and then slowly declining again. Time = 1000, food is 10.02kg/km2*

Set the two species to be equal sizes again. Now change the model so that the new mouse is better at finding food than the old one ie increase the relevant parameter by 50%.

When you’ve done that, run the model and check it is working correctly.

A graph of a mouse

Description automatically generated

\* What happens to the population of the new mouse after colonisation?

\* What happens to the population of the first mouse after colonisation by the new mouse?

\* What happens to the amount of food available after colonisation by the new mouse?

Keep the parameter value you just changed at the same value ie second mouse 50% better at finding food than the first. Can you now change the amount of food needed by the first species to a value where the two species can co-exist in the long-term?

\* Does such a value exist? If so, what is the value? (Don’t waste too much time on this one; come back to it later if you get stuck.)

Now set the values for the mice back to the way they were at the start ie exactly the same.

Seed production by plants is likely to vary over the year, according to the seasons (rainfall, temperature etc). Model this by setting the amount of food produced in a day to be:

1+sin(time/365\*pi\*2)

When you’ve done that, run the model over six years and check it is working correctly.

*Food production is* ***1+sin(time/365\*pi\*2)***

*If I create an object of time = 1:365*

*and foodprod =* 1+sin(time/365\*pi\*2)

*then plot (foodprod) I get the sin curve*

*A graph of a function

Description automatically generated*

\* What day of the simulation within the first year is the day of maximum food production?

*Day 91 using which.max(foodprod)*

\* After things stabilise after colonisation by the new mouse, what happens to the amount of food available over each year?

*A graph of food availability

Description automatically generated*

\* After things stabilise after colonisation by the new mouse, what happens to the first mouse over each year?

A graph of food availability

Description automatically generated

\* In the long run, over a few hundred years, what happens to the population of the new mouse? Is this a different result than when the food doesn’t vary? Why do you think this is the case?

**With varying food**

A black curve with numbers

Description automatically generated

**Without food variation**

**A graph with numbers and a line

Description automatically generated**

**Part 2: Population Dynamics and Chaos**

Ok, now let’s make things really simple again. We are now considering the population density of a kind of fly. Our time step will be a month. The fly is living in an environment where food is supplied at a constant rate. This means that if the population density of the fly is low, then its population will increase. At higher densities, the growth rate slows, and if it gets too high, then it will run out of food, and crash. The higher it is, the lower it will crash.

We can represent this simply with the following equation:

nt+1= g nt (1- nt/1000)

which looks like this when g=3.



The parameter g represents the maximum growth rate at low densities, and also how hard the population crashes. The height of the top of the peak (750 in this case) is the maximum sustainable population density.

\* How does the maximum sustainable population density depend on the value of g?

*I’m sure there’s some maths thing here*

nt = 500

what’s nt+1

then do some maths

Write some R code that simulates how the density of this population will change over 100 months, using a loop. Set the initial population density to 7 and the growth rate to 2. Run the simulation and plot the population density changing over time. You should find that the population increases exponentially at first, then levels out towards an equilibrium population density of 500.

fly = 7

months = 100

growth = 2

for (time in 2:months){

fly[time] = growth \* fly[time-1] \* (1-fly[time-1]/1000)

}

\* Change the growth rate to 0.2 and run the simulation. What happens now?

A graph with numbers and lines

Description automatically generated

\* Change the growth rate to 0.99 and run the simulation. What happens now?

A graph with a line

Description automatically generated

\* Change the growth rate to 1.5 and run the simulation. What happens now?

A graph with numbers and a line

Description automatically generated

\* Change the growth rate to 2.9 and run the simulation. What happens now?

A graph with numbers and lines

Description automatically generated

\* Change the growth rate to 3 and run the simulation. What happens now?

A graph of a graph

Description automatically generated

\* Change the growth rate to 3.1 and run the simulation. What happens now?

A graph of a graph

Description automatically generated with medium confidence

\* Change the growth rate to 3.5 and run the simulation. What happens now?

A graph of a number of months

Description automatically generated with medium confidence

\* Change the growth rate to 3.55 and run the simulation. What happens now?

A graph of a number of months

Description automatically generated

\* Change the growth rate to 3.6 and run the simulation. What happens now?

A graph of a number of months

Description automatically generated with medium confidence

\* Change the growth rate to 3.8 and run the simulation. What happens now?

A graph showing a number of months

Description automatically generated

\* Change the growth rate to 4 and run the simulation. What happens now?

A graph of a number of months

Description automatically generated

\* Keep the growth rate at 4 and change the initial population density to 750 and run the simulation. What happens now?

A graph with numbers and lines

Description automatically generated

\* Keep the growth rate at 4 and change the initial population density to 750.0000001 and run the simulation. What happens now?

A graph of a graph

Description automatically generated

For many values of g (such as g=4) this system is chaotic, meaning that very small differences in starting conditions quickly become magnified over time (the old butterfly effect). In real systems initial conditions are impossible to measure with perfect accuracy, and the system is subject to external influences that change the system. This means that real chaotic systems are impossible to predict, even if we understand the system very well, know the initial conditions with a high degree of accuracy, and the external influences have very minor effects. Nonetheless, chaotic systems, such as naturally varying populations, can be managed and controlled in clever ways.

\* If g=4, there is an initial population density near 350 from which the population will alternate periodically, returning to its initial population every second month. What is this population density (to 3 decimal places of accuracy). You could work this out using algebra or just by experimenting with different values of density. But since it might take quite a while, you should probably leave it for now and come back to it later if you have time.

# Part 3: Population Dynamics of Bacteria Populations

Ok, now we are going to model some bacteria populations changing over time. Look at the following R code.

This is similar to what we did before. There are two species of bacteria, x and y, which are a photosynthesising autotroph and a heterotroph, respectively.

x <- 2

y <- 2

timestep=0.1

t <- seq(0,24\*60,by=timestep)

for (i in 1:length(t)){

growthrate\_x = 0.02\*(1-y[i])

dx=timestep\*growthrate\_x\*x[i]

growthrate\_y = 0.02\*(x[i]-1)

dy=timestep\*growthrate\_y\*y[i]

x[i+1]=x[i]+dx

y[i+1]=y[i]+dy

}

plot(t,x[-length(x)])

points(t,y[-length(y)],col='red')

Populations are measured in biomass within the incubator – in this case we start with 2kg of each species.

The natural time scale of the model is one minute, but for simulation we break each minute into 10 smaller timesteps (timestep=0.1).

The total simulation time is one day (24\*60 minutes).

The change in biomass of species x and y in one of these time steps is dx and dy, respectively. The growth rate of both species is 2% per minute (0.02).

The autotroph is eaten by the heterotroph, and so its growth rate reduces as the population of the heterotroph increases (1-y[i])).

The heterotroph depends on eating the autotroph, so its growth rate increases as the population of the autotroph increases (x[i]-1). Make sure you understand each line of the code.

Copy the code into your R script and run it. Make sure it works and make sure you understand what is happening and why.

Plot the population of the autotroph over time (on the x-axis) against the population of the heterotroph (on the y-axis) and make sure you understand the plot you get.

A black line with white text

Description automatically generated

\* In the plot you get, you should see some kind of loop or circle or spiral - which way round the loop or circle or spiral are the populations moving as time moves forward – clockwise or anticlockwise?

*Clockwise*

\* At the moment the populations are changing continuously, but there are equilibrium populations that will remain constant. One boring answer is that both populations will remain constant if both are equal to zero. What is a more interesting answer? (Maybe leave for later if you get stuck here.)

At the moment, the only thing keeping the population of the autotroph limited is getting eaten by the other species. Check this by setting the initial density of the heterotroph to zero and running the simulation.

\* What happens?

*Unrestricted autotrophic growth*

*A line with numbers and symbols

Description automatically generated*

More realistically, there would be some self-competition for the autotroph too; as population increases there would be more shading that would slow further growth. We can include this in the model by specifying that the growth rate of the autotroph reduces with its own density as well as that of the heterotroph that eats it. Modify the code so that the growth rate modifier is 1-x[i]/10-y[i] instead of just 1-y[i]. Set the initial density of the heterotroph to zero and run the simulation.

\* What happens?

*Autotroph growth is limited – previously the growth rate would’ve been exponential, whereas now there is a steady growth rate.*

A graph with a line

Description automatically generated

Now set the initial population of the heterotroph to be very small eg. 0.01, and run the simulation.

\* What happens?

*I think this is moving in a counter-clockwise direction, as x starts at density 2 and as it increases so too does the density of y before x starts to drop as y increases, and then the cycle continues.*

\* Plot the population of the autotroph against the population of the heterotroph and make sure you understand the plot you get.

A diagram of a curve

Description automatically generated

\* Again at the moment the populations are changing continuously, but again there are equilibrium populations that will remain constant. Once again, one boring answer is that both populations will remain constant if both are equal to zero. What is a more interesting answer? (Maybe leave for later if you get stuck here.)

*Doesn’t work:*

|  |  |  |
| --- | --- | --- |
| *X* | *Y* | *Stable* |
| *2* | *0.2* | *N* |
| *0.2* | *2* | *N (just reverses plot)* |
| *1* | *0.1* | *N (X grows more compared with 2 and 0.2)* |
| *1* | *0.5* | *N* |
| *1* | *1.2* | *N* |
| *10* | *100* | *Getting closer? = see plot* |

Now we add a third species, species Z. This species eats the second one (the old heterotroph, species Y), and has no interaction with the first one (the autotroph, species X). It has the same growth rate of 2% as the other species, and its growth rate slows to zero when the biomass of its food is 1kg, just like the old heterotroph. Add lines of code to represent the new species. The growth rate of species Y will also be affected since it is getting eaten now; the modifier x[i]-1 should be changed to x[i]-1-z[i]. Set the initial population of all three species to be equal to 2.

Run the simulation, check the new code works and that you understand the output.

\* What happens?

A graph with a line

Description automatically generated

\* Plot the population of the autotroph against the population of the old heterotroph and make sure you understand the plot you get. It might help to set the size of the points to represent the biomass of the third species, like this : plot(x,y,cex=z/max(z))

A diagram of a diagram

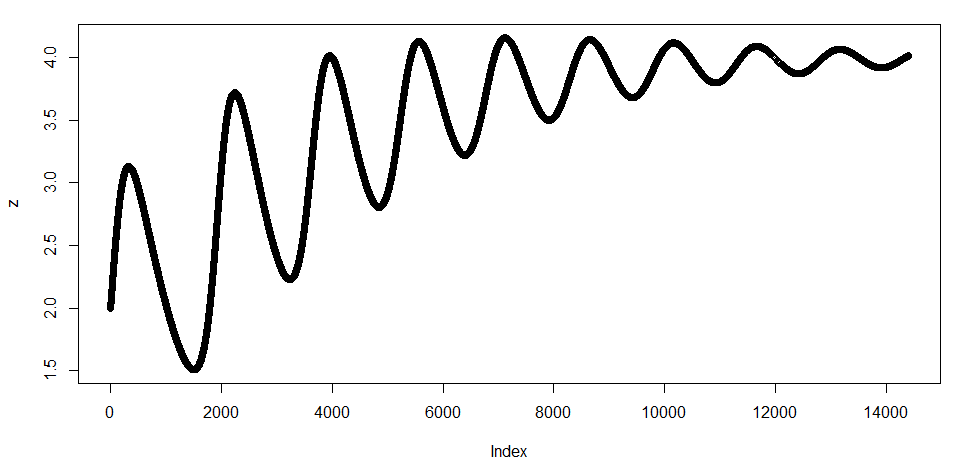
Description automatically generated with medium confidence

\* Again at the moment the populations are changing continuously, but again there are equilibrium populations that will remain constant. Once again, one boring answer is that all three populations will remain constant if all three are equal to zero. What is a more interesting answer? (Maybe leave for later if you get stuck here.)

Now suppose that the new heterotroph (species Z) doesn’t need as much food as our old heterotroph, so its growth rate reaches zero only when the population of species Y is 0.5, instead of 1. Change the relevant number in the model.

Run the simulation, check the new code works and that you understand the output.

\* What happens?



A graph of a function

Description automatically generated

\* Again at the moment the populations are changing continuously, but again there are equilibrium populations that will remain constant. Once again, one boring answer is that all three populations will remain constant if all three are equal to zero. What is a more interesting answer? (Maybe leave for later if you get stuck here.)

All the models so far have assumed a spatially homogenous population ie a population (or populations) that is the same everywhere in space, or least possible differences across space are not accounted for – now we are going to make a spatially explicit population dynamics model ie one that explicitly accounts for variability across space as well as time.

# Part 4: Spatially-Explicit Population Dynamics

Get the ‘parasites1.R’ file from LMS and save it to your own drive, in the same folder you have been using for the other modelling. Make sure that you have this folder set us your working directory in R. Open the file from within R. Run the whole script (you can select ‘Run all’ from the ‘Edit’ menu.)

You should see some pretty colours changing with time. This plot is a coloured spatial representation of a growing and spreading population, where white represents areas of high population density and red areas of low density. We start with a single small area in the middle (at the point (6,6) in fact) where population density is relatively high (200 to start with), and elsewhere the density is zero. Gradually the species increases its density within the area where it began, and at the same time it also starts to spread to adjacent areas. So the model is similar to the previous models of population growth, but with space and spread included too.

This could represent the density of an invasive species like our mouse, or blackberry weed, cane toads or an insect pest that is building up population from an initial small incursion and spreading out across the landscape…

Run this a couple of times and make sure you understand what this is representing.

Look at the code in the file. The first few lines define parameter values.

\*What happens if we change the ‘Hmr’ parameter to 1 instead of 0.1? or to 0? What is this controlling?

*Host migrations/ or rate of migration*

*At 1 the migration reaches the edge of the world by time = 10*

*when 0 the host stays in the same box within the world/neighbourhood? But increases in density*

Change Hmr back to 0.1.

We now introduce a second species that is a parasite or predator of the first species we have been looking at (which is therefore a host species for the parasite or the prey if it is a predator – we’ll refer to them as parasite and host from now on). This could be the insect parasite that attacks the mouse, or a rust fungus that infects the blackberry weed, or a new virus that attacks cane toads, or a little introduced predator wasp that eats the insect pest. We might be introducing this parasite as a form of bio-control.

We introduce 10 parasites at the same starting point (6,6) as the hosts. Do this by finding the second line of code under ##starting population

P[6,6] <- 0 ## 0 parasite individuals at 6,6

and change the 0 to 10 so that it now looks like

P[6,6] <- 10 ## 10 parasite individuals at 6,6

\*Run the model again. NOTE - you cannot see the density of the parasite in the model output, just the effect that it has on the density of the host. What is happening here?

*Parasite inhibits migration rate of the host and decreases the density of the host species*

Now change the ‘pauseTime’ parameter from 900000 to 0, and run again. What happens?

*I have absolutely no idea what happened*

Change the ‘worldsize’ to 100 and run again. What happens?

*Makes world bigger – 6,6 is now in the bottom left corner and tiny*

Change ‘totaltime’ to 200 and run again. What happens now?

*Instead of time 1-10 graphs like before its now generating 200 graphs/maps with the growth of the host species*

*Note: because 6,6 is in the corner with world size of 100 when growth goes off the corner of the map it comes in the other corner*

*Looks like when population density of the host reaches a certain threshold the parasites must have a population boom because they reduce the density of the host species considerably following that threshold limit. Results in beautiful expanding pattern with varying densities across the map.*

**Notice that the world ‘wraps around on itself’** – anything that spreads out of the edge of the world is assumed to spread back in on the opposite side. This is equivalent to assuming that the area we can see and everything that is happening in it is replicated above, below, and to the sides.

Leave these parameters at these new values.

The model outputs densities of the two species densities changing over time at a single point (the starting point, (6,6)) to a file names ‘results.csv’ that you should find in your folder. You could open this file in Excel and make a plot of the two species changing over time, but we can also just do it directly with these lines of code in R:

**plot(as.numeric(record[2:dim(record)[1],1]),as.numeric(record[2:dim(record)[1],2]),t='l',xlab='time',ylab='density')**

**lines(as.numeric(record[2:dim(record)[1],1]), as.numeric(record[2:dim(record)[1],3]), col='blue')**

NOTE – this shows densities changing over time but at one spatial location only!! It’s like we are measuring population density each day in just one particular small area as individuals reproduce, die, and move in and out of this area…

\*What pattern do you see in the dynamics of the population densities at this point? Why?

*As host population density increases parasite density increases, resulting in reduced fitness and decline in host population density. As host population density decreases, parasite load decreases. With reduced parasite load, host population can recover => increased parasite load.*

*As population density increases contact between individuals increases resulting in increased transmission of parasites through vectors/other transmission methods.*

A graph of a person

Description automatically generated

We will now introduce a small second population of the same host species into our ‘world’. Do this by adding the line

N[60,50] <- 2

just below the line

N[6,6] <- 200 ##200 host individuals at 6,6

Also change the starting point of the parasite by changing the line of code

P[6,6] <- 10

to

P[60,50] <- 10

Change the total time to 400. Then run the model again. Look carefully at the dynamic output for the spatial patterns and also the plot of population density at the starting point over time on both a log and standard scale. (You can make an R plot use a log scale on the y-axis by adding the argument log=’y’ to the plot function. You should also add one to the y-variable because otherwise, if you have zeros, the result will be undefined when R logs them for the plot.)

*Might need some help here with this part*

Assume we have one of the bio-control examples from above and write a few sentences explaining what is happening in general terms to population densities of the two species over time and space. Try to explain why this behaviour is occurring. How do you think this model might be used to help with the bio-control program?

*If the parasite is the bio-control agent and it is added to the population at location [60,50] then it restricts both host population migration and density considerably. Whereas, host population [6,6] has no parasite load, and without other modelled limitations this population undergoes continuous growth and spread outwards until it comes into contact with the parasite loaded population. Upon contact, the parasite quickly spreads through the unburdened population [origin 6,6] due to the intense population density. Subsequently, host population [6,6] experiences considerable population collapse as the parasite spreads through. Following parasite introduction the population density of the host species undergoes fluctuations across the world map as the parasite load increases and decreases in a continuous feedback loop.*