

Multi-species interactions

The predator-prey Lotka-Volterra model

Implementation of the base Lotka-Volterra model

During the practical about population dynamics, we saw how to use the `ode()` function from the `deSolve` R package to solve a differential equation for one species. To do so, we had to implement an R function representing the differential equation representing logistic growth:

$$\frac{dP}{dt} = r \left(1 - \frac{P}{K} \right) P$$

```
LG <- function(t, state, parameters){ ##logistic grown function,
that takes a set of parameter values, initial conditions and a
time sequence
  with(as.list(c(state, parameters)),{ ##"with" is a function
that allows us to use the variable names directly - it looks
for r, K and P in state and parameters

    dP <- r*(1-P/K)*P ##this is our logistic equation
governing the rate of change of P

    return(list(dP)) ## return the rate of change - it needs
to be a list
  }) # end with(as.list ...)
}
```

Modify this function to include two differential equations representing a predator-prey Lotka-Volterra model (the two equations must be included within the same function):

$$\frac{dx}{dt} = \alpha x - \beta xy \quad \frac{dy}{dt} = \delta xy - \gamma y$$

Where x is the prey population, y is the predator population, α is the growth rate of the prey population, β is the effect of predators on the prey growth rate, δ is the effect of preys on the predator growth rate, and γ is the predator death rate. Let's call this function `LV()`.

Using the `ode()` function, like last time, we will solve the predator-prey Lotka-Volterra model. You will fill the following code to solve this system of differential equations:

```
state <- ... ## the initial population values
parameters <- ... ## the equation parameters
times <- ... ##a sequence of time steps - uses function seq()
out <- ode(y= ..., times = ..., func = ..., parms = ...)
out.df <- data.frame(out)
```

Fill the code for 500 timesteps with intervals of 0.01 and for parameters:

$\alpha = 0.1$
 $\beta = 0.02$
 $\delta = 0.02$
 $\gamma = 0.4$
 $x(0) = 10$
 $y(0) = 10$

You can now plot the output as:

```
ggplot(data = out.df) +
  geom_line(mapping=aes(x=time,y=X),color="blue") +
  geom_line(mapping=aes(x=time,y=Y),color="red") +
  geom_hline(yintercept=0,color="darkgrey") +
  geom_vline(xintercept=0,color="darkgrey") +
  labs(x = "Time", y = "P")
```

This gives the changes in the two populations through time. As we saw during the lecture, it is also useful to plot the changes in populations through time in the phase space, i.e. to plot one predator population as a function of the prey population. To do so, we can use the following code:

```
ggplot(data = out.df) +
  geom_path(mapping=aes(x=X,y=Y),color="red") +
  xlim(0,70) +
  ylim(0,40) +
  geom_hline(yintercept=0,color="darkgrey") +
  geom_vline(xintercept=0,color="darkgrey") +
  labs(x = "Prey", y = "Predator")
```

Change the values of the parameters and the initial conditions, what happens to the outputs when you increase or decrease them? Discuss.

Prey growth rate: exponential vs logistic

In the original Lotka-Volterra model, the prey growth rate is exponential in the absence of predator. Indeed, if $y = 0$, the equation governing prey growth becomes:

$$\frac{dx}{dt} = \alpha x$$

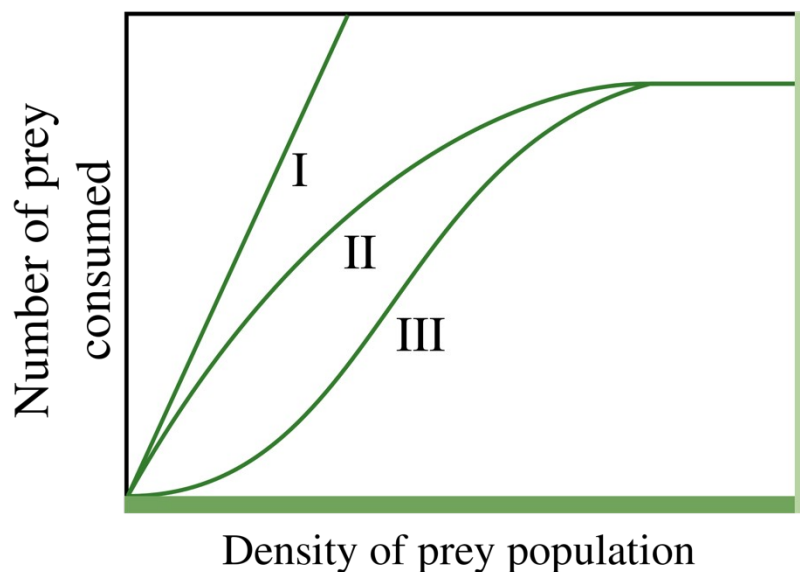
Let's change the equation to include a logistic growth instead:

$$\frac{dx}{dt} = \alpha x \left(1 - \frac{x}{K}\right)$$

Let's set K at 30, and implement this in your system of differential equations in R (i.e. in your function `LV()`). How does it change the results for the same set of parameter values listed above? Discuss (i.e. try to figure out why).

Incorporating functional response

Similar to the difference between exponential and logistic growth, as we saw during the lecture, prey consumption by predators is not always linear. At some point, there will be so many preys that predators will not be able to hunt more than they already are. The rate at which predators can consume preys is called a functional response. There are three common types of functional response, including the linear one (type I) that is captured by the terms βxy and γxy in the original Lotka-Volterra equations.



The three common types of functional response.

A type II functional response can be modelled by the following equations:

$$\frac{dx}{dt} = \alpha x - \frac{\beta xy}{1 + A \times x} \quad \frac{dy}{dt} = \frac{\delta xy}{1 + A \times x} - \gamma y$$

Where A controls the slope of the functional response. A high value of A indicates poor hunting efficiency by the predator. Identify and plot the functional response from these equations for different values of A (i.e. plot a type II functional response as in the figure above) using the following code:

```
x <- seq(0,50,0.1)
A <- ... ###test values here
y <- x/(1+A*x)
ggplot()+
  geom_line(mapping=aes(x=x,y=x/(1+A*x)),color="blue") +
  geom_hline(yintercept=0,color="darkgrey") +
  geom_vline(xintercept=0,color="darkgrey") +
  labs(x = "Prey population", y = "Prey consumed")
```

Include a type II functional response in your implementation of the Lotka-Volterra model in R. How does it change the outputs for different values of A (use the curves plotted with the code above to decide on a range of values)? Discuss.

Even for small values of A, which means the predator is quite efficient, its population will collapse, because a small decrease in predation will limit its birth rate. What do you see in the model output? Does it make sense? Discuss.

This way of modelling functional response is very phenomenological. In other word, we directly represent functional response, without really understanding what causes it. Think about the individual-based predator-prey (wolf-sheep) model seen in class, in which predators and preys move randomly in space. What could generate a functional response in this model?

Combine both the logistic growth and the functional response in your Lotka-Volterra model. How does it change the outputs for different values of A? Discuss.

A three-species competition Lotka-Volterra model: limiting similarity

Let's change the predator-prey Lotka-Volterra model for a competition Lotka-Volterra model, which can be modelled by the following equations:

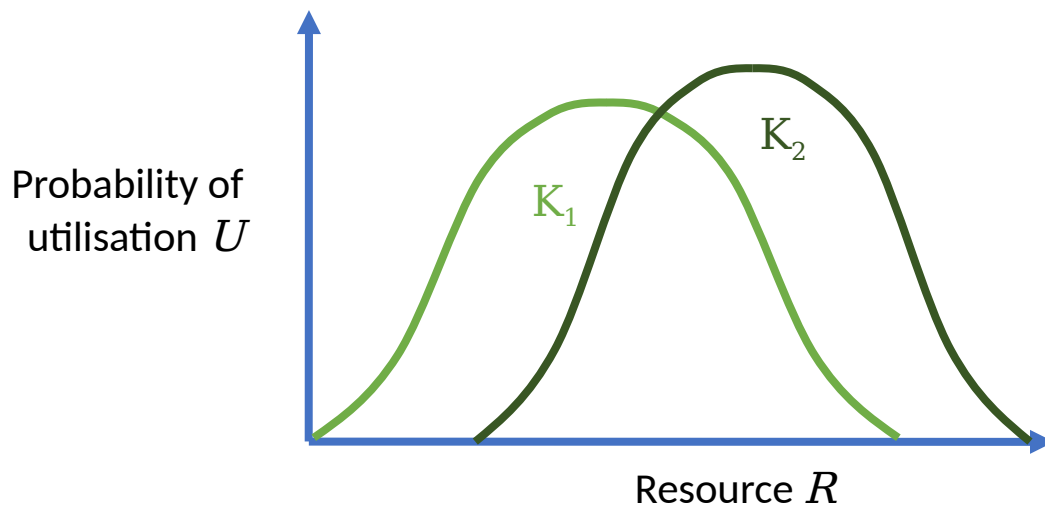
$$\frac{dx_1}{dt} = r_1 x_1 \left(1 - \frac{x_1}{K_1} - \frac{\alpha_{12} x_2}{K_1} \right) \quad \frac{dx_2}{dt} = r_2 x_2 \left(1 - \frac{x_2}{K_2} - \frac{\alpha_{21} x_1}{K_2} \right)$$

Implement this system of differential equations in R, and use the `ode()` function to solve it (you can use the following parameters: $r=0.3$, $x_1(0) = 50$, $x_2(0) = 10$, $\alpha_{12} = 1$, $\alpha_{21} = 0.9$). We have already played a bit with the parameter values in class, but you can do it here with your model, to make sure you get the same results.

Now, we will add a third species:

$$\begin{aligned} \frac{dx_1}{dt} &= r_1 x_1 \left(1 - \frac{x_1 + \alpha_{12} x_2 + \alpha_{13} x_3}{K_1} \right) & \frac{dx_2}{dt} &= r_2 x_2 \left(1 - \frac{x_2 + \alpha_{21} x_1 + \alpha_{23} x_3}{K_2} \right) \\ \frac{dx_3}{dt} &= r_3 x_3 \left(1 - \frac{x_3 + \alpha_{31} x_1 + \alpha_{32} x_2}{K_3} \right) \end{aligned}$$

The α_{ij} parameters will be determined by the relative niches of the species. As seen during the lecture, if I have species 1 and 2 with a Gaussian niche, α_{12} and α_{21} are computed as:



$$\alpha_{12} = \frac{\int U_1(R) \times U_2(R) dR}{\int U_1(R) \times U_1(R) dR}$$

$$\alpha_{21} = \frac{\int U_1(R) \times U_2(R) dR}{\int U_2(R) \times U_2(R) dR}$$

Here is the code to do implement this, you can just copy-paste it:

```
alpha.func <- function(mu1,sig1,mu2,sig2,K1,K2,start,end)
{ ##this is the function to compute the alpha coefficients
  from the mean and standard deviations of the Gaussian niches
  of the species and the start and end values of the environment
  niche1 <- K1 *
  dnorm(seq(start,end,length.out=100),mean=mu1,sd=sig1)
  ##dnorm() generates the values of the Gaussian. Check ?dnorm
  niche2 <- K2 *
  dnorm(seq(start,end,length.out=100),mean=mu2,sd=sig2)
  a <- sum(niche1*niche2)/sum(niche1*niche1) ##because we have
  discrete values, we use a sum to approximate the integral
  return(a)
}
```

```
##Let's try different parameter values
D <- 5 ##distance between the niche optima
mu1 <- 10 ##niche optima of species 1
mu2 <- mu1+D ##niche optima of species 2
mu3 <- mu1+2*D ##niche optima of species 3
sig1 <- sig2 <- sig3 <- 10 ##all species niches have the same
standard deviation for simplicity
K1 <- 200 ##carrying capacity species 1 and 3
K2 <- 250 ##carrying capacity species 2
start <- 0
end <- 30
a12 <- alpha.func(mu1,sig1,mu2,sig2,K1,K2,start,end)
a13 <- alpha.func(mu1,sig1,mu3,sig3,K1,K1,start,end)
```

```

a21 <- alpha.func(mu2,sig2,mu1,sig1,K2,K1,start,end)
a23 <- alpha.func(mu2,sig2,mu3,sig3,K2,K1,start,end)
a31 <- alpha.func(mu3,sig3,mu1,sig1,K1,K1,start,end)
a32 <- alpha.func(mu3,sig3,mu2,sig2,K1,K2,start,end)

##visualise the niches
resource <- seq(start,end,length.out=100)
niche1 <- dnorm(resource,mean=mu1,sd=sig1)*K1
niche2 <- dnorm(resource,mean=mu2,sd=sig2)*K2
niche3 <- dnorm(resource,mean=mu3,sd=sig3)*K1
ggplot()+
  geom_line(mapping=aes(x=resource,y=niche1),color="blue")+
  geom_line(mapping=aes(x=resource,y=niche2),color="red")+

geom_line(mapping=aes(x=resource,y=niche3),color="darkgreen")

##setup and solve the system of differential equations
parameters <- c(a12=a12, a13=a13, a21=a21, a23=a23, a31=a31,
a32=a32, r=0.3, K1 = K1, K2 = K2)
state <- c(X1=10, X2=10, X3=10)

LS2 <- function(t,state,parameters){
  with(as.list(c(state, parameters)),{
    # rate of change
    ##you need to complete this

    # return the rate of change
    list() ##you need to complete this
  }) # end with(as.list ...)
}

times <- seq(0,200,by=0.01)
out <- ode(y=state, times = times, func = LS2, parms =
parameters)
out.df <- data.frame(out)

##plot the populations
ggplot(data = out.df)+
  geom_line(mapping=aes(x=time,y=X1),color="blue") +
  geom_line(mapping=aes(x=time,y=X2),color="red") +
  geom_line(mapping=aes(x=time,y=X3),color="darkgreen") +
  geom_hline(yintercept=0,color="darkgrey") +
  geom_vline(xintercept=0,color="darkgrey") +
  labs(x = "Time", y = "P")

```

Take some time to look at the code, to understand what is happening. With these parameters, which species survive? Why? Can you find two other combinations of parameters generating two different outputs? Explain why these combinations of parameters generate these outputs (check the slides from the lecture).

That was too easy and you have some extra time?

Implement a type III functional response in the predator-prey Lotka-Volterra model and explore how it changes the outputs compared to type I and II functional responses. The type III functional response can be modelled by the following equations:

$$\frac{dx}{dt} = \alpha x - \frac{\beta x^2 y}{1 + A \times x^2} \quad \frac{dy}{dt} = \frac{\delta x^2 y}{1 + A \times x^2} - \gamma y$$

Do the same things combining the different types of functional responses with logistic instead of exponential growth.

Draw the vector field for a set of parameter values for a Lotka-Volterra model.

- Use for loops to iterate through all (x,y) values
- Compute the values of dx/dt and dy/dt according to your equations
- Use function `arrows()` to plot the arrows showing the vector field