|  |
| --- |
| Department of Ecology |
|  |
|  |

Describing population dynamics of experimental data using a prey-predator model

Pascal Bärtschi

.

Table of Contents

[1 Introduction 3](#_Toc106575423)

[2 Methods 4](#_Toc106575424)

[2.1 Data processing 4](#_Toc106575425)

[2.2 Simulation of the Lotka Volterra System 4](#_Toc106575426)

[2.2.1 System properties 4](#_Toc106575427)

[2.2.1.1 Parameters of prey equation 4](#_Toc106575428)

[2.2.1.2 Parameters of predator equation 4](#_Toc106575429)

[2.2.1.3 Assumptions 4](#_Toc106575430)

[2.2.2 System dynamics 5](#_Toc106575431)

[2.2.3 Implementation of the simulation 5](#_Toc106575432)

[2.2.3.1 Core Functions 5](#_Toc106575433)

[2.2.3.2 Workflow of simulation 5](#_Toc106575434)

[2.3 Fitting the simulation 6](#_Toc106575435)

[2.3.1 Ordinary Least Squares 6](#_Toc106575436)

[2.3.2 Nelder-Mead optimization 6](#_Toc106575437)

[2.4 Analyzing the fitted simulation 6](#_Toc106575438)

[3 Results 8](#_Toc106575439)

[3.1 Simulation versus experimental data 8](#_Toc106575440)

[3.2 Simulations aim for stability 8](#_Toc106575441)

[3.2.1 Exponential decay of cycle tops 9](#_Toc106575442)

[3.2.2 Number of Oscillations 10](#_Toc106575443)

[4 Discussion, conclusion & outlook 11](#_Toc106575444)

[4.1 Wide variety of fulfilled assumptions 11](#_Toc106575445)

[4.2 Density effect 11](#_Toc106575446)

[4.3 Effect of short generation time 11](#_Toc106575447)

[4.4 Conclusion: additive effect 11](#_Toc106575448)

[4.5 Alternative Approach in Insects 12](#_Toc106575449)

[4.6 Outlook 12](#_Toc106575450)

[5 Self-assessment 13](#_Toc106575451)

[6 References 13](#_Toc106575452)

[7 Appendix 13](#_Toc106575453)

# Introduction

Models describing population dynamics rank among the most important tools of ecology. After making observations of a population’s growth behaviour and its interaction with the environment, ecologic models are used to quantify the observed behaviour. Models help us to understand the systems mechanism by forcing us to formulate observed processes and assumptions. Consequences of these mechanisms can then be shown. Even general predictions about population size or density are made possible. Therefore, ecologic models help us understanding and explaining reality by simplifying the interactions in the frame of suitable assumptions.



Figure 1: Planktonic rotifer grazing on unicellular algae. Scale: 102 µm. Credit: Bob Blaylock at English Wikipedia

The cyclic behaviour of prey-predator systems is one of the most fundamental concepts in population dynamics as they occur in a large range of communities in the wild and under experimental conditions. The cyclic dynamics arise from trophic interactions between predator and prey. These dynamics were described by Alfred J. Lotka (1925) and Vito Volterra (1926) independently by a pair of differential equations.

The core of this project is to describe a predator-prey dynamic observed in the field by this Lotka-Volterra system of equations. The chosen ecosystem consisted of two interacting, cultured freshwater organisms. The data was collected by Blasius et. al1 in an experimental environment with stable condition to investigate the long-term persistence of cyclic predator-prey systems. The predators of this artificial ecosystem are planktonic rotifers, which grazed on unicellular algae being their prey. The used open-access dataset contained density data of prey and predator occurrence over a period of 357 days.

To describe the predator-prey dynamic, the Lotka-Volterra model was fitted to the observed data with help of the Nelder-Mead optimization algorithm with the ordinary-least-squares (OLS) method being the objective function.

The goal of this project was to minimize the OLS distance between my implemented Lotka-Volterra simulation and the dataset by optimizing the non-linear optimization problem through the Nelder-Mead algorithm. The cyclic dynamic of this artificial ecosystem would then be described by the Lotka-Volterra equation parameter, which would facilitate the understanding of the trophic interaction between planktonic rotifers and unicellular algae. Leading us to the research question: How good can the Lotka-Volterra model describe data observed in the field?

# Methods

## Data processing

At the beginning of the project the open-access dataset has been saved as a csv-file2 from figshare.com. The file has then been loaded into a jupyter notebook as a pandas data frame. All rows containing NA values have been dropped. The column containing the algae density was multiplied by a hundred to change the scale from 106 cells/ml to 104 cells/ml creating a working initial input into the model. As a last step the data was cut down to the columns needed for the project: the timeline, algae- and rotifer-density. To simplify data loading the steps above have been implemented into a function.

## Simulation of the Lotka Volterra System

### Shape, arrow Description automatically generatedSystem properties

Figure 2: Flowchart of predator-prey equation

Consisting of two classes, the predator y and the prey x class, the system is described by a total of four parameters in two differential equations. One equation is allocated to the change per time of prey population size, being referred to as the prey equation. Describing the same in change of predator, the predator equation completes the system.

Equation 1: Prey equation

Equation 2: Predator-Equation

#### Parameters of prey equation

The exponential growth of the prey is represented by the term *αx* under the assumption that prey has an unlimited supply and reproduces exponentially. The interaction term *βxy* describes the predation upon the prey, assumed to be proportional to the rate prey and predator meet.

#### Parameters of predator equation

In the predator equation *δxy* represents the growth of the predator population. Note that this rate is not necessarily the same as the predation rate of prey, because the predator population does not grow with the same rate as it consumes the prey due to losses in metabolism. The loss of predators is described by the term *γy,* being an exponential decay if the prey is absent.

#### Assumptions

In addition to the assumptions referred to above the model assumes that there is only interaction between the predator and the prey. Thus, the predator feeds uniquely on one prey species and the prey is hunted unquely by the predator species. Furthermore, the described ecosystem is assumed to have constant biotic and abiotic factors, excluding dynamics such as apparent competition or trophic cascades. On top the ecosystem is assumed to be closed, excluding immigration or emigration of organisms.

The resulting simplicity of the assumptions made is one hand an advantage of this model by describing a complex dynamic by few variables and on the other hand a disadvantage, because simplicity fails to map reality precise in most cases.

### Chart, line chart Description automatically generatedSystem dynamics

Figure 3: Cyclic oscillation of the prey in green followed by the predator in red.

Text

Description automatically generated with medium confidenceDiagram, shape, polygon

Description automatically generated with medium confidence

Figure 4: The four phases rotating counter clockwise. In the example of Figure 3 the dynamic starts at the top left phase

The trophic interactions of predator and prey and the resulting changes in population densities lead to repeated cyclic oscillations over time (Figure 3). Each oscillation consists of four phases seen in Figure 4, which rotate counter-clockwise. The system dynamic seen in Figure 4 starts with the phase at the top left corner, where prey first decrease and predators follow due to rare prey. Afterwards, the prey density increases due to low predation while the predators are still decreasing. Predator density now starts to increase caused by high prey abundance. In the last phase the prey density starts decreasing again due to high predation rate, followed by the predators decreasing again in the next oscillation. Note that the isoclines in Figure 4 are not density dependent of prey nor predator, which is another assumption of the Lotka-Volterra model gaining simplicity.

The resulting dynamic is the prey density peaks preceding the peaks of predator as result of the predator-prey interaction mechanism. This shift in time of the peaks is referred to as the time-lag and differs from prey-predator pairs to others.

### Implementation of the simulation

#### Core Functions

The prey-predator system has been simulated by two functions defined in a jupyter notebook. The first function contains the pair of Lotka-Volterra equations and returns dx/dt and dy/dt in a numpy-array. To iterate over this function, I have chosen the Runge-Kutta forth order method, which is implemented in the second function. The fourth order Runge-Kutta method (Equation 5) has been chosen due to showing the most precise results in one of the assessments of BIO 394. The Runge-Kutta function returns a two-dimensional array with each dimension being a population density. The length of the returned array is representative for the number of time steps put into the function.

Equation 3: Forth order runge-kutta equation as core of differential equation solving function.

#### Workflow of simulation

The interaction workflow of the function goes as follows. A pandas time series, the initial predator-prey density, the pair of differential equations (Lotka-Volterra system) and the parameters for the equations are put into the Runge-Kutta function. The number of iterations equals the length of the time series minus one to avoid boundary problems. In every iteration the delta of the timepoint of the past iteration to the timepoint of the running iteration is calculated. This is important because the populations in the used dataset haven’t been observed perfectly regular, which results in different time periods between observations. The Runga-Kutta function then uses the Lotka-Volterra function to calculate dx/dt and dy/dt to seen in Equation 3 as *dfdt*. At the end of the ith iteration the array is put into the ith place of the returned final array.

## Fitting the simulation

### Ordinary Least Squares

Chart

Description automatically generated

Figure 5: Visualisation of the OLS distance (red), the simulation (black) and the datapoints (blue).

To optimize the distance between the datapoints calculated by the simulation and the points provided by the dataset an objective function is needed. I have chosen the ordinary least sum of squares (OLS) to measure this distance, because it is the method I already know from previous solved linear regression problems. It seems reliable, simple and straight forward fitting the frame of the Lotka-Volterra model. I have implemented it into a function in my notebook which takes two arrays of the same length as an input and then returns the calculated OLS distance between them.

### Nelder-Mead optimization

The goal of the fitting process is to minimize this objective function by using the Lotka-Volterra parameter as hyperparamters put into an optimization algorithm. Hereby the simulated datapoints are shifted closer to observed ones of the dataset

The Nelder-Mead optimization method is chosen to minimize the OLS distance. This algorithm is a numerical method used to minimize an objective function in non-linear minimization problems. Nelder-Mead is chosen, because it needs few function evaluations per iterations which leads to the algorithm being little time consuming.

The method is also referred to as the downhill simplex method which perfectly describes its workflow. The algorithm first needs an initial guess, which should be relatively close to the deserved output to avoid ending up in a local minimum of the density map. To achieve this optimal initial guess the simulations behaviour has been investigated manually by visualization of different parameter combinations.

Chart, surface chart

Description automatically generated

Figure 6: Iterations of Nelder-Mead shown as simpex "rolling down" the density map. .Lighter colors show are more advanced iteration.

The Nelder-Mead method is simplex based, which means that the initial parameter guess is stored as simplex. N being the number of parameters, the working simplex has n+1 vertices leading to working in n dimensions. N dimensions consists of the parameter valued x-axes and the y-axis represents the computed objective function value. One having n+1 vertices leads to n+1 parameter test points around the initial guess.

In every function evaluation the value of the objective function at every test point is extrapolated. The test point with the highest function value is then substituted by a better one lying “inside” the simplex. This way the simplex “rolls” down the density map into a local or global minimum, as seen in Figure 6.

The process is terminated by the simplex becoming sufficiently small returning the test point with the smallest function value.

The project applied the algorithm by using the minimize function from scipy optimize in python. The OLS function was put in as a lambda function and important to note is that tight boundaries for the axis limits had to be set for a successful process termination

## Analyzing the fitted simulation

After fitting the simulation with Nelder-Mead, different approaches and tools are used to analyze it. To plot the simulated and experimental data matplotlib in python is used as well as pandas functions such as rolling() are used to emphasize strength and weaknesses of the simulation.

Finally, an exponential decrease of the cycling tops has been modelled. To achieve a fitting exponential decay, the cycle tops had to be found first. This is made through a self-implemented function with a rolling window looking for the maximum inside this window. The window size has been manually set.

To fit an exponential function through these tops curve\_fit() from scipy optimize is used. This function executes the Levenberg-Marquart algorithm, which is used for non-linear least squares problems. This makes it suitable for curve fitting. The challenge here is again to find an initial guess which leads to an optimal fit and thus avoids ending up in a local minimum.

# Results

## Chart, histogram Description automatically generatedSimulation versus experimental data

Figure 7: Simulation dynamci in bold versus experimental data shown transparent in background.

Minimizing the OLS score between the simulation and the with the best suited initial guess led to the system dynamic seen in figure 7. After successful termination Nelder-Mead algorithm returned an OLS score of 612’146 and the parameter combination α= 0.043, β= 0.003, γ= 0.417 and δ= 0.007 in 240 iterations with a total of 406 function evaluations.

Chart, line chart

Description automatically generatedGoing by visual inspection of figure 7 the oscillation extent of the simulation and the data match. Furthermore, the algae densities increase before they are followed by the increase of rotifers. However, the dynamics of the simulation do not seem represent the one observed in the experiment.

Figure 8: Simulation over 1000 days plotted with its rolling mean +/- rolling standard deviation.

There are far too few cycles over the period of 357 days in the simulation. The system in the experiment shows about 50 prey-predator cycles in comparison to the simulation which only oscillated four times. Additionally, the simulated system seems to decrease in oscillation strength with increasing time.

The experimental system shows stability of prey-predator cycle extend over time, whereas the optimized simulation seems to aim for a stable equilibrium with dx/dt = dy/dt = 0. A stable equilibrium is not seen in predator prey systems, where cycle dynamic persist unless one of the species emigrates or dies out.

## Simulations aim for stability

To further investigate the poorly fitting dynamics of the simulation, it was simulated over a period of 1000 days.

This enforces the suspected effects of decreasing cycle stability and decreasing oscillation strength.

In figure 8 one see that the rolling means do not change much over time, supporting the hypothesis of the aiming for stable equilibrium. However, the standard deviation decreases 14.83 units over algae and 12.79 units over the rotifer cycles. This infers the decrease of cycle Chart, radar chart

Description automatically generatedoscillation strength.

Figure 9: Prey against predator density gets closer to overall. One Rotation equals one oscillation shown in Figure 7.

Figure 9 compares the algae plotted against the rotifer density of the experiment data with the simulation for 1000 days. While the simulation gets closer to its overall mean with every cycle, the experimentally observed dynamic show a more stable cyclic structure with few extreme peaks.

However, these extreme peaks have not been expected to occur the simulation because no randomness was included.

### Chart, line chart Description automatically generatedExponential decay of cycle tops

Figure 10: Exponential decrease of cycle tops over time.

The decrease of the oscillation tops over time suggests an exponential decay, like most decays observed in biological dynamics. As seen in figure 10 the decay of both cycle top arrays could be predicated by an exponential function.

Equation 4: Exponential decay of algae tops with time

Equation 5: Exponential decay of rotifer tops with time.

The algae oscillation tops are decaying slightly faster than the rotifer tops (Equation 4 vs. Equation 5). To achieve this result, it is again important to give the function an input close to the tops to avoid getting stuck in local minima.

### Number of Oscillations

Chart

Description automatically generated with medium confidence

Figure 11: Above: Initial guess with lowest OLS score showing stable cycles over time. Below: initial guess with relatively short cycle duration reaching equilibrium quickly.

Lastly the number of oscillation cycles being too little has been investigated. A wide variety of initial parameter guesses for the Nelder-Mead optimization method has been tried out. The result of this exploration has not led to a better fit or a dynamic with more cycles. Instead, a trade-off between the initial guesses’ cycle number per time and the stability of the cycles over time.

In figure 11, one can see that the lower initial guess shows skyrocket-like behaviour in the first two oscillations, but then reaches a stable equilibrium quickly.

The upper dynamic is the initial guess that leads to the lowest OLS score shown in figure 11, showing stable cyclic behaviour.

# Discussion, conclusion & outlook

## Wide variety of fulfilled assumptions

At first sight the Lotka-Volterra model seems almost perfect to describe the dynamics of the experimental system.

Chart, histogram

Description automatically generatedThe exponential growth assumption for prey having unlimited resource supply is fulfilled by the nutrient concentration held constant in the system. There are also only the algae and the rotifer interacting which leads to the rotifer uniquely feeding on one prey. Thus, the abiotic and biotic factor were held constant making other dynamics such as apparent competition or trophic cascades impossible, fulfilling another assumption.

Furthermore, the ecosystem in the experimental system was closed. Fulfilling all these assumptions at a time is rare in the wild and can only be achieved in experimental studies. It seemed like THE chance for the Lotka-Volterra model to prove that simplicity can model complex ecologic population dynamics.

## Density effect

However, the model assumes that the organisms’ growth and their interaction is not density dependent. Thus, the predators’ effect on the prey is independent of prey density. This assumption is clearly violated.

Diagram

Description automatically generatedPer-capita are effects in prey lead to self-limitation if packed to dense. This means that the loss of prey is not solely dependent on the described interaction term *βax* in the model, but also on the prey density. In the same way predators limit themselves if abundant to dense.

Figure 12: Density dependence of prey on the left and of predators on the right.

The divergence of the density dependent in figure 12 compared to the non-dependent effect in figure 4 is crucial.

## Effect of short generation time

Another important aspect to investigate is the short generation time of the two freshwater microorganisms. The Lotka-Volterra model was developed doing research on vertebrate species. Such species have generation times lying in the range of years whereas microorganism can double through vegetative reproduction within hours. This divergence has strong effect on the time-lag of the different cycle dynamics, leading to diverging durations of one cycle.

Figure 13: Cycle dynamics of a vertebrate predator-prey system.

As an example, we see the predator-prey dynamic of the snowshoe hare and the Canadian lynx in figure This vertebrate dynamic oscillated 9 times in 90 years leading to a cycle period of 10 years. Compared to the microorganism having a duration of roughly seven days by oscillating about 50 times in 357 days.

## Conclusion: additive effect

The reason for the Lotka-Volterra model to failing relfect the dynamic off the microorganism lies in the combination of the unincluded density dependence and the short generation time.

The system being density dependent is important due to two aspects.Firstly, the predation on the algae differs due to its density. If unicellular algae are very rare ,the rotifer has to search longer per prey as when they are clustered on one hand . On the other hand it gets difficult for the predator to reach many alage if they are together in big clusters. The rotifer can only graze on the algae lying on the outside, the algae on the inside is protected by the surrounding individuals. Secondly, one can see the discussed self-limatiation of predator and prey if becoming to abundant as seen in figure 12.

In additon the model is not able to handle the short generation times combined with relative stable cycles over time as seen in figure 7. The model was simply not built to mirror a dynamic of increasing and decreasing populations as quickly as we see it.

The two violated assumption add up in the Lotka-Volterra Model failing to reflect the data.

## Alternative Approach in Insects

An approach to solve this problem is used in parasite(=predator)-prey relationships in insects. Insects have generation sizes in the week to month range. As a timeline the generations instead of the time that has passed was taken. This way the actual influence from the trophic interaction from one generation to next was measured. This makes the Lotka-Volterra model parameters more meaningful, because i.e. growth rates make sense to measure from one generation to the next. Unfortunately this approach would be too difficult to use in microorganims because the generations are difficult to differ when in interaction with others.

## Outlook

Thus, the chosen model doesn’t suit to model the dynamics. To continue on this process I would either try building a model of differential equations myself or use an agent based approach.

With all the new knowledge of interaction and density factor gathered so far, building a model myself will be easier than at the beginning of this project.

# Self-assessment

Working on this project for about two months taught me things the past two years of my bachelor couldn’t. For the very first time I had to come up with my own research question and find I way to answer it through the language of code.

The way I found was not always an easy one, which made it a lot harder to accept that the research question had to be answered with a ‘The model cannot describe the observed data at all.’. In the end I learned that even a project failing to fulfil the goal can be very useful, because one searches for reasons and explanations for the failure.

I think the goal of this project was achieved in some way, because I managed to fit the model through the way I wanted to. I implemented the simulation, optimized the OLS score with Nelder-Mead and analyzed the results and their explanation in a variety of plots. Only the outcome of the simulation fit could’ve been more satisfying.

If I could start the project from the beginning with the knowledge I have now, I would be more careful creating the projects frame. This time I first chose to model I wanted to do research with and searched for a fitting dataset afterwards. This procedure is wrong from the ground up, because in ecology one first observes a dynamic and implements a method to describe and explain it as the second step. With this lesson learned I am ready to jump into my next research project!

# References

Footers:

1. Long-term cyclic persistence in an experimental predator–prey system, Blasius et. Al
2. Dataset C1: <https://figshare.com/articles/dataset/Time_series_of_long-term_experimental_predator-prey_cycles/10045976/1>

Figures:

* Figure 1: <https://en.wikipedia.org/wiki/Rotifer#/media/File:Bdelloid_Rotifer.jpg>
* Figure 3,4 &12: Ökologie kompakt (Wolfgang Nentwig, Sven Bacher, Roland Brandl)
* Figure 6: <https://edwardwebster.me/2020/04/26/c-coursework/> (\*\*modified)

Sources:

* <https://en.wikipedia.org/wiki/Lotka%E2%80%93Volterra_equations>
* <https://studyflix.de/biologie/lotka-volterra-regeln-2468>
* https://en.wikipedia.org/wiki/Nelder%E2%80%93Mead\_method
* Ökologie kompakt (Wolfgang Nentwig, Sven Bacher, Roland Brandl)

# Appendix

OneDrive Link of notebooks:

* https://1drv.ms/u/s!AqpdZ-tkeBi1nhCZjvMaWdicp7p5?e=f1WGhx