Fitting an ecologic model to field data

# 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.



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 was 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. al 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 dynamic would than be described by the Lotka-Volterra equation parameter, which would facilitate the understanding of the trophic interaction between planktonic rotifer and unicellular algae.Thus, the research question is: 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 was saved as a csv-file 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 to make the model work. As a last step the data was cut down to the columns needed for the project: the timeline, algae-density 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

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.

Text

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

In addition to the assumptions referred to above the model assumes that there is only interaction between the predator and the prey. It is assumed that the predator feeds uniquely on this prey species and that the prey is hunted only 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.

Chart, line chart

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

### System dynamics

Diagram, shape, polygon

Description automatically generated with medium confidenceThe trophic interactions of predator and prey and the resulting changes in population densities lead to repeated cyclic oscillations over time (Figure 4). Each oscillation consists of four phases seen in Figure 5, 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 5 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

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

Text

Description automatically generated with medium confidenceThe interaction workflow of the function went as follows. A pandas time series, the initial predator-prey density, the pair of differential equations (LV-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. This is important because the populations in the used dataset weren’t observed perfectly regular which resulted in different time periods between observations. The Runga-Kutta function then used the Lotka-Volterra function to calculate dx/dt and dy/dt to seen in Figure. At the end of the ith iteration the array was put into the ith place of the later returned array.

## Fitting the simulation

Chart

Description automatically generatedTo optimize the distance between the datapoints calculated by the simulation and the points provided by the dataset an objective function was needed. I chose the ordinary least sum of squares (OLS) to measure this distance, because it was the method I already knew from previous solved linear regression problems. It seemed reliable, simple and straight forward fitting the frame of the Lotka-Volterra model. I implemented it into a function in my notebook which took two arrays of the same length as an input and then returned the calculated OLS between them.

The goal of the fitting process was 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 was 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 was 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 was investigated manually by visualisation of different parameter combinations. 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. 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 where lighter colour show more advanced iterations. 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

Chart, surface chart

Description automatically generated

## Analyzing the fitted simulation

After fitting the simulation with Nelder-Mead different approaches and tools are used to analyse it. To plot the simulated and experimental data matplotlib in python was 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 has been 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 has been used. This function executes the Levenberg-Marquart algorithm, which is used for non-linear least squares problems. This makes it suitable for curve fitting. A challenge here was again to find an initial guess which lead to an optimal fit and thus avoids ending up in a local minimum.

# Results

Chart, histogram

Description automatically generated

Minimizing the OLS score between the simulation and the with the best suited initial guess led to the system dynamic seen in Figure. 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. Going by visual inspection of Figure 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. There are far to few cycles over the period of 357 days in the simulation. The system in the experiment showed 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 system, where cycle dynamic persist unless one of the species emigrates or dies out.

Chart, line chart

Description automatically generated

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 to rolling mean surrounded by the standard deviation of simulated datapoints were plotted over the systems dynamics. 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 oscillation strength.

Figure 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 few extreme peaks. However, these extreme peaks were not expected by the simulation because no randomness was included.

Chart, radar chart

Description automatically generated

Chart, line chart

Description automatically generated

The decrease of the oscillation tops over time suggests an exponential decay like most decays observed in biological dynamics. As seen in figure the decay of both cycle top arrays could be predicated by an exponential function. The algae oscillation tops are decaying slightly faster than the rotifer tops. Top achieve this result it was again important to give the function an input close to the tops to avoid getting stuck in local minima. For minimization of the function curve\_fit from scipy optimize performing the Levenberg-Marquart method.

equations

Chart

Description automatically generated with medium confidence

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

# Discussion, conclusion & outlook

At first sight the Lotka-Volterra model seems almost perfect to describe the dynamics of the expermimental system. The exponential growth assumption for prey having unlimited resource supply is fulfilled by the nutrient concentration held constant in the system. There were also only the algae and the rotifer interacting which led to the rotifer uniquely on one prey. Thus, the abiotic and biotic factor were held constant making other dynamics such as apparent competition or trophic cascades impossible and 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.

Diagram

Description automatically generated

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

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. 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.

Chart, histogram

Description automatically generatedChart, line chart

Description automatically generated

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.

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

Thus, the chosen model doesn’t suit to model the dynamics. To continue on this process I would either try to 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

* Rotifer/algae: [https://upload.wikimedia.org/wikipedia/commons/thumb/2/2d/Bdelloid\_Rotifer.jpg/197px-Bdelloid\_Rotifer.jpg 17/06/2022](https://upload.wikimedia.org/wikipedia/commons/thumb/2/2d/Bdelloid_Rotifer.jpg/197px-Bdelloid_Rotifer.jpg%2017/06/2022), 4pm
* <https://en.wikipedia.org/wiki/Lotka%E2%80%93Volterra_equations>
* https://studyflix.de/biologie/lotka-volterra-regeln-2468
* Ökologie kompakt: modified

# Appendix