Fast fitting of neural ordinary differential equations by Bayesian neural gradient matching to infer ecological interactions from time series data

Willem Bonnaffé[1,2]
Supervised by Ben Sheldon[1] & Tim Coulson[2]

1. Edward Grey Institute of Field Ornithology, Department of Zoology, Oxford University, Zoology Research and Administration Building, 11a Mansfield Road, Oxford OX1 3SZ

2. Ecological and Evolutionary Dynamics Lab, Department of Zoology, Oxford University, Zoology Research and Administration Building, 11a Mansfield Road, Oxford OX1 3SZ

Emails: willem.bonnaffe@stx.ox.ac.uk; tim.coulson@zoo.ox.ac.uk; ben.sheldon@zoo.ox.ac.uk;

Running title:

Keywords: Artificial Neural Networks; Ecological Dynamics; Ecological interactions; Geber Method; Neural Ordinary Differential Equations; Ordinary Differential Equations; Prey-predator dynamics; Time series analysis; Rotifers; Microcosm;

Specifications: xxx words in abstract; xxxx words in text; xx references; 6 figures; 2 tables

Contact: Willem Bonnaffé, 61 St Giles, Pusey House, St Cross College, Oxford, OX1 3LZ, UK (w.bonnaffe@gmail.com)

Statement of authorship:

Abstract

1 Introduction

2 Material and Methods

2.1 Method overview

We aim to provide a non-parametric method for estimating ecological interactions from time series data of species density. We do this by approximating the dynamics of each species with neural ordinary differential equations (NODEs, Bonnaffé, Sheldon, and Coulson 2021). We then compute ecological interactions as the sensitivity of these dynamics to a change in the respective species densities. We provide a novel method, *Bayesian neural gradient matching*, which results in over 300 time faster NODEs fitting.

2.2 Neural ordinary differential equation

A NODE is a class of ordinary differential equation (ODE) that is partly or entirely defined as an artificial neural network (ANN). They are useful to infer dynamical processes non-parametrically from time series data (Bonnaffé, Sheldon, and Coulson 2021). We choose NODEs over standard statistical approaches because they offer two advantages. The first is that NODEs approximate the dynamics of populations non-parametrically. NODEs are therefore not subjected to incorrect model specifications (Jost and Ellner 2000; Adamson and Morozov 2013). This provides a more objective estimation of the inter-dependences between state variables. The second advantage is that it is a dynamical systems approach. So that the approach includes lag effects through interacting state variables, not only direct effects between them.

We first consider a general NODE system,

$$\frac{dy_i}{dt} = f_p\left(y, \theta_i\right), \tag{1}$$

where dy_i/dt denotes the temporal change in the i^{th} variable of the system, y_i, as a function of the other state variables $y = \{y_1, y_2, ..., y_I\}$. The function f_p is a non-parametric function of the state variables and its shape is controlled by the parameter vector θ_i. In the context of NODEs, non-parametric functions are ANNs. The most common class of ANN used in NODEs are single-layer fully connected feedforward ANNs (e.g. Wu, Fukuhara, and Takeda 2005), also referred to by single layer perceptrons (SLPs, Bonnaffé, Sheldon, and Coulson 2021),

$$f_p\left(y, \theta_i\right) = f_\lambda\left(\theta_i^{(0)} + \sum_{j=1}^{J} \theta_{ij}^{(1)} f_\sigma\left(\theta_{ij}^{(2)} + \sum_{k=1}^{K} \theta_{ijk}^{(3)} y_k\right)\right), \tag{2}$$

which feature a single layer, containing J neurons, that maps the inputs, here the state variables y, to a single output, the dynamics of state variable i, dy_i/dt. The parameter vector θ_i contains the weights $\theta^{(l)}$ of the connections in the SLPs. SLPs can be viewed as weighted sums of activation functions f_σ, which are usually chosen to be *sigmoid* functions $f(x) = 1/(1+\exp(-x))$. The link function f_λ allows to map the output of the network to a specific domain, for instance applying *tanh* will constrain the dynamics between -1 and 1, $dy_i/dt \in \ ]-1, 1[$.

We would like to stress that this general form can be changed to represent biological constraints on the state variables. In particular for population dynamics, the state variables are strictly positive population densities, $y_i = N_i \in \mathscr{R}^+$. We could hence re-write equation (1) as, $dN_i/dt = f_p(N, \theta_i)N_i$, where the SLPs approximate the per-capita growth rate of the populations. More details regarding these models can be found in our previous work (Bonnaffé, Sheldon, and Coulson 2021).

2.3 Fitting NODEs by Bayesian neural gradient matching

In this section, we describe how to estimate the parameters θ of the NODE system given a set of time series. Fitting NODEs can be highly computationally intensive, which hinders uncertainty quantification, cross-validation, and model selection (Bonnaffé, Sheldon, and Coulson 2021). We solve this issue by introducing *Bayesian neural gradient matching* (BNGM), a computationally efficient approach to fit NODEs. The approach involves two steps (Fig. 1). First, we interpolate the state variables and their dynamics with neural networks (Fig. 1, red boxes). Second, we train each NODE to satisfy the interpolated state and dynamics (Fig. 1, blue boxes). This bypasses the costly numerical integration of the NODE system and provides a fully mathematically tractable expression for the posterior distribution of the parameter vector θ. We coin the term BNGM to emphasise two important refinements of the standard gradient matching algorithm. The first is that we use neural networks as interpolation functions, and the second is that we use Bayesian regularisation to limit overfitting and estimate uncertainty around parameters.

Interpolating the time series

The first step is to interpolate the time series and differentiate it with respect to time in order to approximate the state and dynamics of the variables. We perform the interpolation via non-parametric regression of the interpolating functions on the time series data,

$$Y_{it} = \tilde{y}_i(t, \omega_i) + \varepsilon_{it}^{(o)}, \tag{3}$$

where Y_{it} is observed value of the state variable i at time t, $\tilde{y}_i(t, \omega_i)$ is the value predicted by the interpolation function given the parameter vector ω_i, and $\varepsilon_{it}^{(o)}$ is the observation error between the observation and prediction. The interpolation function is chosen to be a neural network,

$$\tilde{y}_i(t, \omega_i) = f_\lambda \left(\omega_i^{(0)} + \sum_{j=1}^{J} \omega_{ij}^{(1)} f_\sigma \left(\omega_{ij}^{(2)} + \omega_{ij}^{(3)} t \right) \right), \tag{4}$$

where the parameter vector ω_i contains the weights $\omega^{(l)}$ of the network. We can further differentiate this expression with respect to time to obtain an interpolation of the dynamics of the state variables (Fig. 1, red boxes),

$$\frac{d\tilde{y}_i}{dt}(t, \omega_i) = \sum_{j=1}^{J} \omega_{ij}^{(1)} \omega_{ij}^{(3)} \frac{\partial f_\sigma}{\partial t} \left(\omega_{ij}^{(2)} + \omega_{ij}^{(3)} t \right) \frac{\partial f_\lambda}{\partial t} \left(\omega_i^{(0)} + \sum_{k=1}^{J} \omega_{ik}^{(1)} f_\sigma \left(\omega_{ik}^{(2)} + \omega_{ik}^{(3)} t \right) \right). \tag{5}$$

Fitting NODEs to the interpolated time series

The second step is to train the NODE system (Eq. 1) to satisfy the interpolated dynamics. Thanks to the interpolation step, this simply amounts to performing a non-parametric regression of each NODE (Eq. 1) on the interpolated dynamics (Eq. 5),

$$\frac{\partial \tilde{y}_i}{\partial t}(t, \omega_i) = \frac{dy_i}{dt}(\tilde{y}, \theta_i) + \varepsilon_{it}^{(p)}, \tag{6}$$

where $\varepsilon_{it}^{(p)}$ is the process error, namely the difference between the interpolated dynamics, $\partial \tilde{y}_i / \partial t$ and the NODE, dy_i/dt, given the interpolated state variables $\tilde{y} = \{\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_I\}$ (Fig. 1, blue boxes).

Bayesian regularisation

In the context of standard gradient matching, defining the observation model (Eq. 3) and process model (Eq. 6) would be sufficient to fit the NODE system (Eq. 1) to the time series via optimisation. We could find the parameter vector ω_i and θ_i that minimise the sum of squared observation and process errors, $\varepsilon_{it}^{(o)}$ and $\varepsilon_{it}^{(p)}$ (Eq. 3 and 6). However, this approach is prone to overfitting, and does not provide estimates of uncertainty around model predictions. To account for this, we introduce Bayesian regularisation, which allows us to control for overfitting by constraining parameters with prior distributions (Cawley and Talbot 2007), and to root our interpretation of uncertainty in a statistically sound framework.

First, we define a simple Bayesian model to fit the interpolation functions (Eq. 3) to the time series data. We assume normal distributions for the observation error, $\varepsilon_{ij}^{(o)} \sim \mathcal{N}(0, \sigma_i)$, and for the parameters, $\omega_{ij} \sim \mathcal{N}(0, \gamma_{ij})$. Here, we are only interested in interpolating the time series accurately, irrespective of the value of σ_i and γ_{ij}. Therefore, we use the approach developed by Cawley and Talbot to average out the value of the parameters σ_i and γ_{ij} in the full posterior distribution (Cawley and Talbot 2007), assuming gamma hyperpriors $p(\xi) \propto \frac{1}{\xi} \exp\{-\xi\}$ for both parameters. This yields the following expression for the log marginal posterior density of the parameters,

$$\log P(\omega_i \mid Y_i) \propto -\frac{J}{2} \log \left(1 + \sum_{j=1}^{J} \left(\varepsilon_{ij}^{(o)} \right)^2 \right) - \frac{K}{2} \log \left(1 + \sum_{k=1}^{K} \omega_{ik}^2 \right) \tag{7}$$

where P is the marginal posterior density, $\omega_i = \{\omega_{i1}, \omega_{i2}, ..., \omega_{iK}\}$ is the observation parameter vector controlling the interpolation function, $Y_i = \{Y_{i1}, Y_{i2}, ..., Y_{iJ}\}$ corresponds to the sequence of observations of state variable i at time step j, J is the total number of time steps in the time series, $\varepsilon_{ij}^{(o)}$ is the observation error at time step j between the interpolated and observed value of variable i, K is the total number of parameters. More details on how to derive this expression can be found in a supplementary file (Supplementary A).

Then, we define a simple Bayesian model to fit the NODEs to the interpolated dynamics, given the interpolated states. We assume normal distributions for the observation error, $\varepsilon_{ij}^{(p)} \sim \mathcal{N}(0, \sigma_i)$, and parameters, $\theta_{ik} \sim \mathcal{N}(0, \delta_{ik})$. This gives the following expression for the log posterior density of the parameters given the interpolations,

$$\log p(\theta_i \mid \omega) \propto -\frac{1}{2} \sum_{J=1}^{J} \left(\frac{\varepsilon_{ij}^{(p)}}{\sigma_i} \right)^2 - \frac{1}{2} \sum_{k=1}^{K} \left(\frac{\theta_{ik}}{\delta_{ik}} \right)^2 \tag{8}$$

where $\theta_i = \{\theta_{i1}, \theta_{i2}, ..., \theta_{iK}\}$ are the NODE parameters of the i^{th} variable, $\omega = \{\omega_1, \omega_2, ..., \omega_I\}$ are the interpolation parameters of each state variable, $\varepsilon_{ij}^{(p)}$ is the process error of variable i at time step j between the interpolated dynamics and NODE prediction, σ_i is the standard deviation of the likelihood, K is the total number of parameters, δ_{ik} is the standard deviation of the prior distribution of parameter θ_{ik}.

This approach allows us to limit overfitting by adjusting the constraint on the parameters, which is controlled by the standard deviation of the parameter prior distributions, δ_{ik} (Cawley and Talbot 2007; Bonnaffé, Sheldon, and Coulson 2021). We could set small values of δ to limit the degree of non-linearity in the response, or to eliminate specific variables from the model by constraining their parameters to be close to zero. We identify the appropriate degree of constraint δ_i on NODE parameters via cross-validation. We train the NODE model on the first half of the interpolated data and predict the remaining half. We repeat this process for increasing values of δ_i, until we find the value that maximises the log likelihood of the test data.

2.4 Inference and uncertainty quantification

Finally, we estimate uncertainty in parameter values by *anchored ensembling*, which produces approximate Bayesian estimates of the posterior distribution of the parameters (Pearce et al. 2018). This involves sampling a parameter vector from the prior distributions, $\theta_i \sim \mathcal{N}(0, \delta_i)$, and then optimising the posterior distribution from this starting point, $\theta_i^* = \underset{\theta_i}{argmax} \log p(\theta_i \mid \omega)$. By repeatedly taking samples, the sampled distribution θ^* approaches the posterior distribution and provides estimates and error around the quantities that can be derived from the models. The expectation and uncertainty around derived quantities can then be obtained by computing the mean and variance of the approximated posterior distributions. The great strength of this approach is that it is unlikely to get stuck in local maxima and provides a more robust optimisation of the posterior.

2.5 Analysing NODEs

In this study we are mainly interested in two outcomes of NODEs, namely inferring the direction (or effect) and strength (or contribution) of interactions between the state variables (Bonnaffé, Sheldon, and Coulson 2021). We define the direction of the interaction between variable y_i and y_j as the derivative of the dynamics of y_i with respect to y_j, and vice versa,

$$e_{ijt} = \frac{\partial}{\partial y_j} \frac{dy_i}{dt}. \tag{9}$$

Knowing the direction, however, is not sufficient to determine the importance of a variable for the dynamics of another. Given the same effects, a variable that fluctuates a lot will have a greater impact on the dynamics of a focal variable, compared to a variable that remains quasi-constant. We hence compute the strength of the interaction by multiplying the dynamics of a variable y_j by its effect on the focal variable y_i, also known as the Geber method (Hairston et al. 2005),

$$c_{ijt} = \frac{dy_j}{dt} \frac{\partial}{\partial y_j} \frac{dy_i}{dt}. \tag{10}$$

To summarise results across the entire time series we can compute the mean effects e_{ij} by averaging e_{ijt} across all time steps, $e_{ij} = K^{-1} \sum_k e_{ijk}$, as well as the relative total contribution, c_{ij}, of a variable to the dynamics of another by computing the relative sum of square contributions, $c_{ij} = \left(\sum_{ijk} c_{ijk}^2\right)^{-1} \sum_t c_{ijt}^2$. By computing the direction and strength of interactions between all the variables in the system we can build dynamically informed ecological interaction networks (e.g. fig. 5). Other metrics can be computed by analysing the NODEs, such as equilibrium states, these are discussed in our previous work (Bonnaffé, Sheldon, and Coulson 2021).

3 Case study 1: artificial tri-trophic prey-predator oscillations

In this first case study, we aim to demonstrate the accuracy of the NODE fitted by BNGM in inferring non-linear per-capita growth rates in a system where ground truth is known. Hence, we simulate a set of time series from a tri-trophic ODE model with known equations and parameters, and we compare the fitted NODEs to the actual ODEs.

3.1 System

We consider a tri-trophic ODE system consisting of a prey, an intermediate predator, and a top predator. The system is built on the real tri-trophic system featuring algae, flagellates, and rotifers, considered in case study 3,

$$\begin{aligned}
\frac{dG}{dt} &= \left(\alpha\left(1-\frac{G}{\kappa}\right) - \frac{\beta B}{1+\delta G} - \frac{\gamma R}{1+\delta G}\right) G \\
\frac{dB}{dt} &= \left(\frac{\beta G}{1+\delta G} - \phi R - \mu\right) B \\
\frac{dR}{dt} &= \left(\frac{\gamma G}{1+\delta G} + \phi B - \nu\right) R,
\end{aligned} \tag{11}$$

where G, B, and R, correspond to the prey, intermediate and top predator population densities, respectively, α is the prey intrinsic growth rate, limited by a carrying capacity κ, β and γ are the predation rates by the intermediate and top predator, δ is the saturation rate of prey predation, which emulates the capacity of the algae to display predator defense at higher algal density (Hiltunen et al. 2013), ϕ is the predation rate of the intermediate predator by the top predator, μ and ν are the intrinsic mortality of the intermediate and top predator.

We simulate a case of invasion, by introducing the top predator from rare, with a set of parameters that result in dampening prey-predator oscillations, namely $\alpha = 1$, $\beta = 2.5$, $\gamma = 1.5$, $\kappa = 3$, $\delta = \phi = \mu = \nu = 1$. We focus on the middle section of the time series, $t \in [20,50]$, as in the initial section the rotifer predator is rare, and in the later section populations have attained a fixed equilibrium point. The resulting time series are presented in figure 2.

3.2 NODE model

In order to learn non-parametrically the per-capita growth rate of each species, and to derive ecological interactions, we define a three-species NODE system,

$$\begin{aligned}\frac{dR}{dt} &= r_R(R,G,B,\beta_R)R \\ \frac{dG}{dt} &= r_G(R,G,B,\beta_G)G \\ \frac{dB}{dt} &= r_B(R,G,B,\beta_B)B,\end{aligned} \tag{12}$$

where the per-capita growth rates r_R, r_G, and r_B are neural network functions of the density R, G, B of each species (function f_p, Eq. 2). We choose a combination of linear and exponential activation functions $f_{\sigma,j\leq J/2}(x) = x$, and $f_{\sigma,j>J/2}(x) = \exp(x)$. This allows us to progressively switch from a simple linear model to a non-linear model by releasing the constraint on the exponential section of the neural network during cross-validation. The number of units in the hidden layer J is chosen to be 10, as this is a commonly used number for systems of that size (e.g. Wu, Fukuhara, and Takeda 2005; Bonnaffé, Sheldon, and Coulson 2021).

3.3 Time series interpolation

We interpolate the time series using the neural network described in section 2.3 (Eq. 4). We set the number of neurons in the network to $J = 30$. We use sinusoid activation functions, $f_\sigma(x) = sin(x)$, so that the weights $\omega_{ij}^{(1)}$, $\omega_{ij}^{(2)}$, and $\omega_{ij}^{(3)}$ control the amplitude, shift, and frequency of the oscillations in the time series, respectively. Given that the population densities are strictly positive $R,\ G,\ B \in \mathscr{R}^+$, we use an exponential link function, $f_\lambda(x) = \exp(x)$. We then approximate the marginal posterior distribution of the interpolation parameters, and thereby of interpolated states and dynamics, by taking 100 samples from the log marginal posterior distribution (Eq. 7) via anchored ensembling. In practice, the high number of parameters in the neural network equation may impede the fit of the time series, especially for small time series. We found that dividing the number of parameters K (Eq. 7) by the number of neurons in the network J (Eq. 2) yields consistent fitting results. Interpolated states and dynamics are presented in figure 2.

3.4 Fitting NODEs to the interpolated time series

We fit the NODE system to the interpolated time series. In practice, we fit the NODE to the expectation of the interpolated state and dynamics, $\mathrm{E}(\tilde{y}_i)$ and $\mathrm{E}(d\tilde{y}_i/dt)$, by averaging over all sampled interpolation parameters. An alternative approach could be to consider the interpolation that maximises the log marginal posterior density, but this may decrease repeatability due to the difficulty of reliably identifying a global maximum. Averaging across multiple interpolations ensures an overall smoother and robust interpolation. In addition, we standardise the response and explanatory variables with respect the their mean and standard deviation (i.e. $Z = (Y - \mu)/\sigma$). This is to facilitate the training of the NODE by equalizing the scale of the different parameters in the neural network. Then, we identify the optimal regularisation parameter δ (Eq. 8) by cross validation. To do that, we split the data in half and calculate the log likelihood of the test set for increasing values of δ, from 0.05 (linear) to 0.5 (highly non-linear), by increments of 0.05. This allows us to identify the maximum degree of non-linearity, δ, in the per-capita growth rate that ensures generalisability throughout the time series. Then, we approximate the posterior distribution of the NODE parameters by taking 30 samples from the posterior distribution (Eq. 8). Finally, we perform model selection by removing variables that do not result in a significant decrease in the log likelihood of the model (assessed by comparing log likelihood confidence intervals). We ensure moderate temporal autocorrelation and normality by visualising the residuals of the models. We also ensure results repeatability running the entire fitting process a second time.

3.5 Computing ecological interactions

Finally, we analyse the shape of the per-capita growth rates to recover the interaction between the three species in the system. In particular, we look at the effect and contribution of each species to the dynamics of the others. The effect is computed as the sensitivity (i.e. the gradient) of the per-capita growth rate of a given species with respect to the density of the other species. The contribution is computed following the Geber method (Hairston et al. 2005), which consists in multiplying the dynamics of a variable by its effects on the other variables. We further compute the importance of a species in driving the dynamics of another by computing its relative total contribution compared to other species. More details on how to compute these quantities can be found in section 2.5 and in our previous study (Bonnaffé, Sheldon, and Coulson 2021).

4 Case study 2: real tri-trophic prey-predator oscillations

In this second case study, we want to assess the quality of the NODE analysis when performed on a real time series. We are further interested in comparing the direction and strength of uncovered ecological interactions across virtually identical replicated time series.

4.1 System

We consider a three-species laboratory microcosm consisting of an algal prey (*Chlorella autrophica*), a flagellate intermediate predator (*Oxyrrhis marina*), and a rotifer top predator (*Brachionus plicatilis*). The algal prey is consumed by the intermediate and top predator, which also consumes the intermediate predator (Arndt 1993). The dynamics of this system, here the daily change in the density of each species, were recorded in three replicated time series experiments performed by Hiltunen and colleagues (Hiltunen et al. 2013). We use their time series because they describe a simple yet biologically realistic ecosystem, and because the quality of the replication of their microcosm reduces as much as possible observational and experimental error, and rules out environmental variation (Hiltunen et al. 2013). We digitised these time series by extracting by hand the coordinates of every points in the referential of the axis of the graph of the original study, and analysed them.

4.2 NODE analysis

We apply the same analysis as performed on the artificial tri-trophic prey-predator oscillations. This allows us to recover a non-parametric approximation of the growth rate of each species, and then derive the direction and strength of the ecological interactions that underpin their dynamics. We present detailed results of the analysis of the first time series (Fig. 4), and a summary comparison of the three time series (Fig. 5).

5 Case study 3: real di-trophic prey-predator oscillations

Finally, we infer ecological interactions by NODE BNGM in the hare-lynx system. This is to provide an example of a longer time series, and to offer a point of comparison with previous and future implementations of NODEs, which commonly use this time series (e.g. Bonnaffé, Sheldon, and Coulson 2021).

5.1 System

The system is described in details in our previous work (Bonnaffé, Sheldon, and Coulson 2021). The data consist in a 90-year long time series of counts of hare and lynx pelts collected by trappers in the Hudson bay area in Canada (Odum and Barrett 1972). The time series displays characteristic 10-year long prey-predator oscillations.

5.2 NODE analysis

We apply the same analysis as previously described, to the exception that the NODE system only features two variables, H and L, instead of 3. Results are presented in figure 6.

6 Results

6.1 Model runtimes

We present a breakdown of the runtime of fitting NODEs by BNGM for each system in table 1. We find that it takes on average 5.35 minutes to fit NODEs by BNGM. This includes taking 390 samples, and thereby performing 390 full optimisations, of the posterior distribution of the interpolation and NODE parameters. This amounts to about 5.37 second to sample each variable of the NODE system once. This is a 335 fold improvement over our previous approach, which took on average 30 minutes (Bonnaffé, Sheldon, and Coulson 2021).

6.2 Case study 1: artifical tri-trophic system

We present the results of fitting NODEs by BNGM to the artificial tri-trophic time series in figure 2 and 3. We find that both the interpolation of the state variables and dynamics are highly accurate (Fig. 2), given that they closely match the ground truth, known from the equations of the ODE model that we used to generate the time series (Eq. 11). Similarly, we find that the NODE approximation of the per-capita growth rate of each species also closely matches the ground truth (Fig. 3, a., d., g.). We find negative non-linear effects of the two predators on the growth rate of the algae (Fig. 3, b., blue and purple lines). This non-linear pattern is mirrored by the effect of the algae on the growth rate of the predators (Fig. 3, e. and h., red line). The linear interaction between the two predators is also well-recovered (Fig. 3, e., blue line, and h., purple line). We find that removing the intra-specific dependence in the growth rate of the predators did not affect the fit of the model (Fig. e., purple line, and h., blue line). The BNGM approach hence recovers accurately the dynamical characteristics of the artificial system.

6.3 Case study 2: real tri-trophic system

First, we present the in-depth analysis of the drivers of the dynamics of the algae, flagellate, and rotifer population in replicate A (Fig. 4). Cross validation reveals that there is no support for non-linear effects in the growth rate of the algae and flagellate for replicate A (Fig. 4, a. and b., d. and e.). We find negative linear intra-specific density-dependence (Fig. 4, b., red line), and negative linear inter-specific effects of the two predators (purple and blue line). We find that the growth rate of the flagellate is virtually solely driven by predation by the rotifer (Fig. 4, e. and f., blue line). The rotifer population itself is driven by a positive non-linear effect of both preys (Fig. h., red and purple line). There is also evidence for positive non-linear intra-specific density-dependence (Fig. h., blue line). Overall, comparing results across the three replicates reveals that the effect of the rotifer population on the flagellate and algae, and the effect of the algae on the rotifer, are the strongest and most consistent interactions (Fig. 5, table 2). The interactions of the flagellate with the algae, and its effect on the rotifer population varies substantially (Fig. 5, table 2). Interestingly, intra-specific density-dependence in rotifer and algae is also found to be inconsistent across the three replicates.

6.4 Case study 3: real di-trophic system

Finally, we present the analysis of the drivers of the hare-lynx population dynamics in figure 6. Cross-validation provides support for non-linear effects in the per-capita growth rate of the hare and lynx. We find that the hare population growth rate is mostly determined by a non-linear negative effect of the lynx population (Fig. 6, b. and c. blue line), and by weak non-linear positive density-dependence (red line). The lynx growth rate is determined by a positive non-linear effect of the hare (Fig. 6, e. and f., red line), and to a lesser extent by negative non-linear intra-specific density-dependence (blue line).

7 Discussion

Acknowledgments

We thank warmly the Ecological and Evolutionary Dynamics Lab and Sheldon Lab Group at the department of Zoology for their feedback and support. We thank Ben Sheldon for insightful suggestions on early versions of the work. The work was supported by the Oxford-Oxitec scholarship and the NERC DTP.

Data accessibility

All data and code will be made fully available at https://github.com/WillemBonnaffe/xxx/xxx.

Statement of authorship

References

Adamson, M. W. and A. Y. Morozov (2013). "When can we trust our model predictions? Unearthing structural sensitivity in biological systems". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 469.2149, pp. 1–19.

Arndt, H. (1993). "Rotifers as predators on components of the microbial web (bacteria, heterotrophic flagellates, ciliates) - a review". In: *Hydrobiologia* 255-256.1, pp. 231–246.

Bonnaffé, W., B. C. Sheldon, and T. Coulson (2021). "Neural ordinary differential equations for ecological and evolutionary time series analysis". In: *Methods in Ecology and Evolution* 2, pp. 1–46.

Cawley, G. C. and N. L. C. Talbot (2007). "Preventing over-fitting during model selection via bayesian regularisation of the hyper-parameters". In: *Journal of Machine Learning Research* 8, pp. 841–861.

Hairston, N. G. J., S. P. Ellner, M. A. Geber, T. Yoshida, and J. A. Fox (2005). "Rapid evolution and the convergence of ecological and evolutionary time". In: *Ecology Letters* 8.10, pp. 1114–1127.

Hiltunen, T., L. E. Jones, S. P. Ellner, and N. G. J. Hairston (2013). "Temporal dynamics of a simple community with intraguild predation: an experimental test". In: *Ecology* 94.4, pp. 773–779.

Jost, C. and S. P. Ellner (2000). “Testing for predator dependence in predator-prey dynamics: A non-parametric approach”. In: *Proceedings of the Royal Society B: Biological Sciences* 267.1453, pp. 1611–1620.

Odum, E. P. and G. W. Barrett (1972). “Fundamentals of Ecology”. In: *The Journal of Wildlife Management* 36.4, p. 1372.

Pearce, T., F. Leibfried, A. Brintrup, M. Zaki, and A. Neely (2018). “Uncertainty in Neural Networks: Approximately Bayesian Ensembling”. In: *arXiv*, pp. 1–10.

Wu, J., M. Fukuhara, and T. Takeda (2005). “Parameter estimation of an ecological system by a neural network with residual minimization training”. In: *Ecological Modelling* 189.3-4, pp. 289–304.

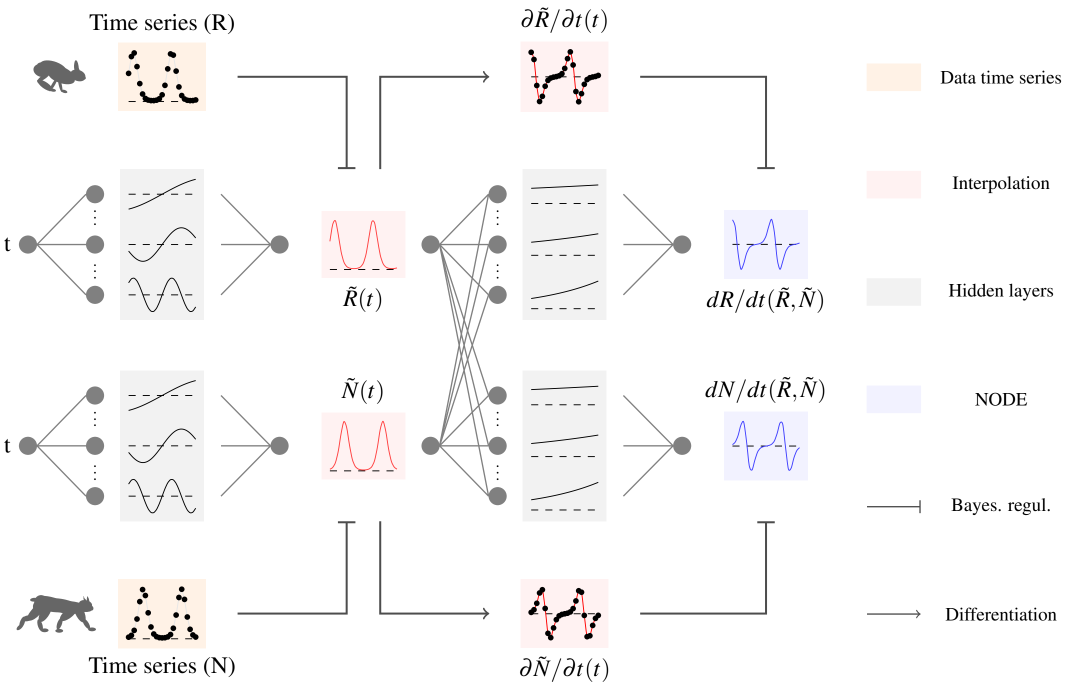

Figure 1: Overview of fitting neural ordinary differential equations (NODE) by Bayesian neural gradient matching (BNGM). In a first step we compute a continuous time approximation (interpolation) of each state variables, here the prey $\tilde{R}(t)$ and predator density $\tilde{N}(t)$. To do that we fit an ANN, that takes time as input, to each time series, via Bayesian regularisation. Interpolated dynamics of populations can then be computed by taking the derivative of the ANN with respect to time, $\partial\tilde{R}/\partial t$ and $\partial\tilde{N}/\partial t$. In a second step, we fit each NODE, dR/dt and dN/dt, to the interpolated dynamics. To do that we fit an ANN, which takes as input the interpolated variables $\tilde{R}(t)$ and $\tilde{N}(t)$, to the interpolated dynamics $\partial\tilde{R}/\partial t$ and $\partial\tilde{N}/\partial t$, via Bayesian regularisation. It takes on average 5.37 seconds to fit NODEs by BNGM, compared to 30 mins in a previous study (Bonnaffé, Sheldon, and Coulson 2021), which corresponds to a 335 fold increase in speed.

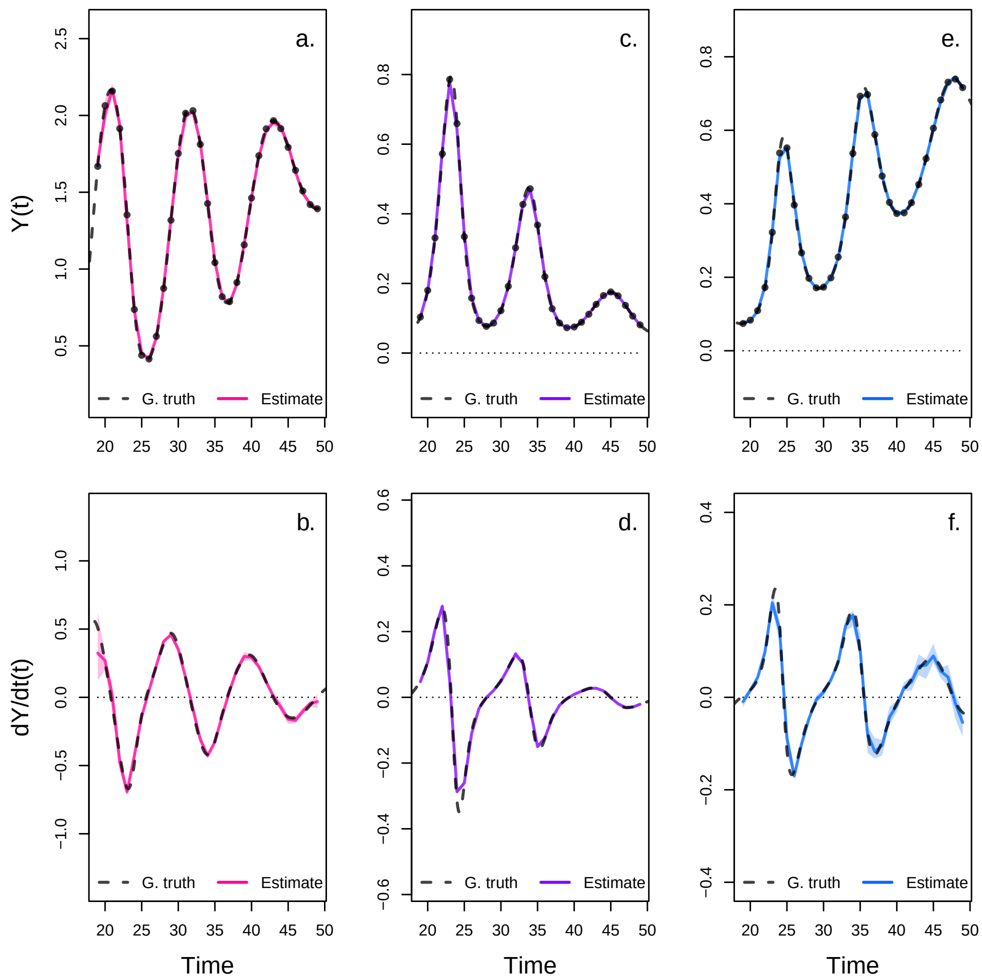

Figure 2: Interpolated density and dynamics of algae, flagellate, and rotifer in the artificial system. This figure corresponds to the first step in the overview figure. It shows the accuracy of the interpolated densities of algae (a.), flagellate (c.), and rotifer (e.). We obtain interpolated densities by fitting observed densities (black dots) with ANNs that take time as input. The observed densities were obtained by sampling a tri-trophic prey-predator ODE model at regular time steps. We then derive interpolated dynamics (b., d., f.) by computing the temporal derivative of the interpolated densities with respect to time. In all graphs, the dashed line represents the ground truth, namely trajectories generated by the ODE model. The solid lines correspond to the interpolations. The shaded area shows the 90% confidence interval, obtained by approximately sampling the marginal posterior distributions.

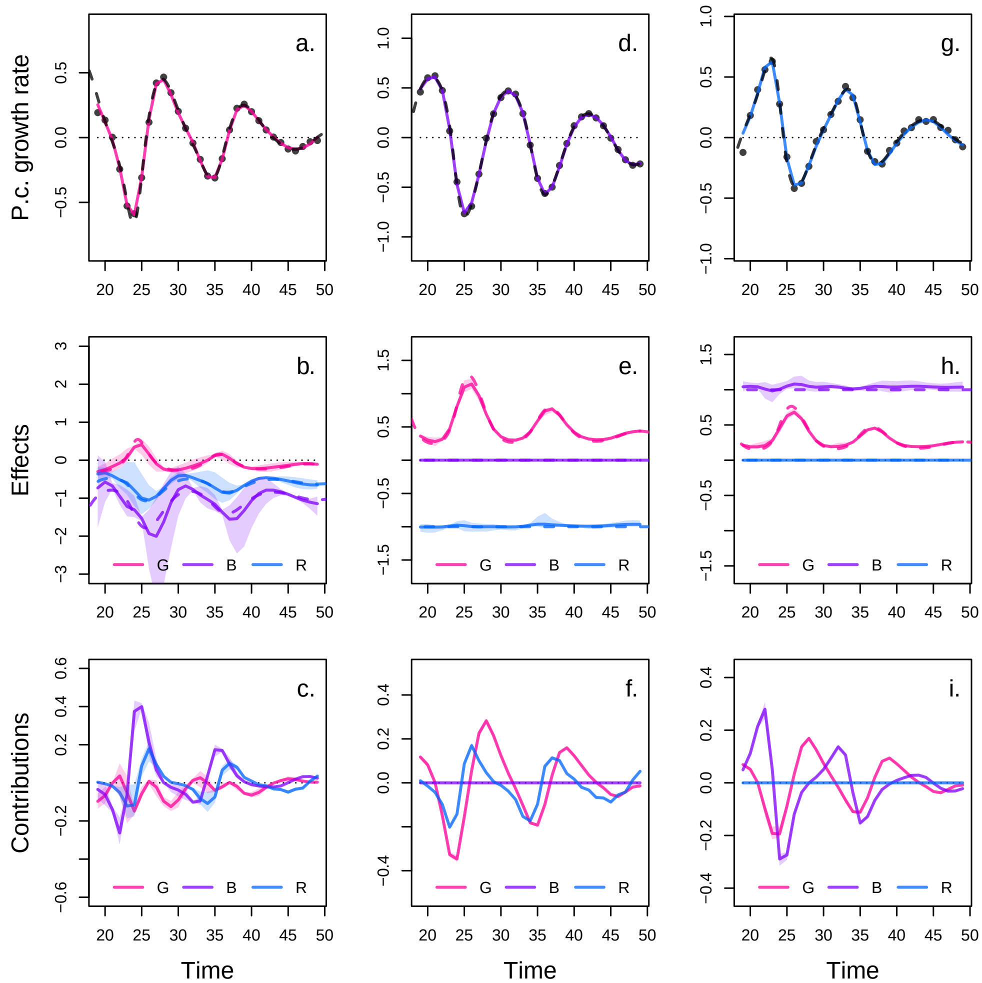

Figure 3: Drivers of dynamics of algae, flagellate, and rotifer in the artificial system. This figure corresponds to the second step in the overview figure. It displays the NODE non-parametric approximations of the per-capita growth rate of algae (a., b., c.), flagellate (d., e., f.), and rotifer (g., h., i.). We obtain the NODE approximations (a., d., g., solid line) by fitting the interpolated per-capita growth rates (black dots) with ANNs that take population densities as input. We then estimate the direction of ecological interactions (effects, b., e., h.) by computing the derivative of the NODE approximations with respect to each density. Finally, we compute the strength of ecological interactions (contributions, c., f., i.) by multiplying the interpolated dynamics of each population (fig. 1, b., d., f.) with its effects. Dashed lines correspond to ground truth, obtained from the original trajectories of the tri-trophic ODE model. The shaded area shows the 90% confidence interval, obtained by approximately sampling the posterior distributions.

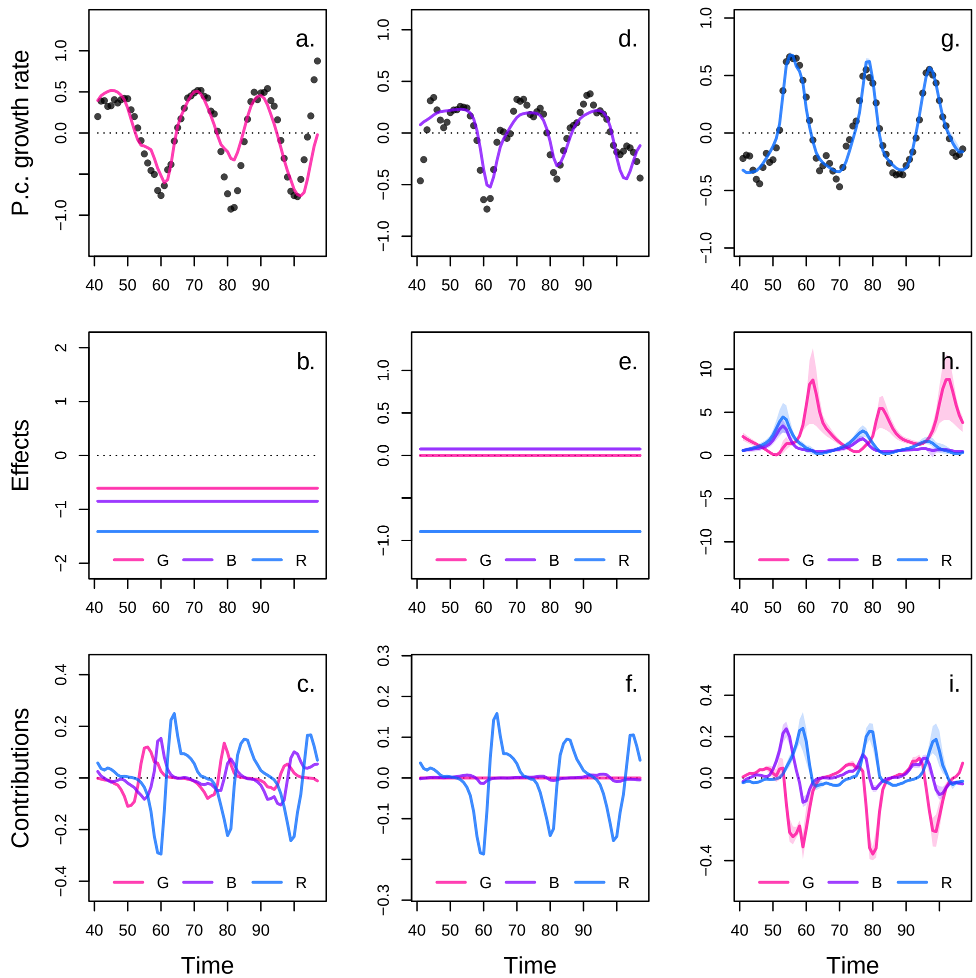

Figure 4: Drivers of dynamics of algae, flagellate, and rotifer in replicate A. This figure displays the NODE non-parametric approximations of the per-capita growth rate of algae (a., b., c.), flagellate (d., e., f.), and rotifer (g., h., i.). We obtain the NODE approximations (a., d., g., solid line) by fitting the interpolated per-capita growth rates (black dots) with ANNs that take population densities as input. We then estimate the direction of ecological interactions (effects, b., e., h.) by computing the derivative of the NODE approximations with respect to each density. Finally, we compute the strength of ecological interactions (contributions, c., f., i.) by multiplying the interpolated dynamics of each population with its effects. The shaded area shows the 90% confidence interval, obtained by approximately sampling the posterior distributions. The replicated time series were obtained by digitising the time series in Hiltunen et al. (2013).

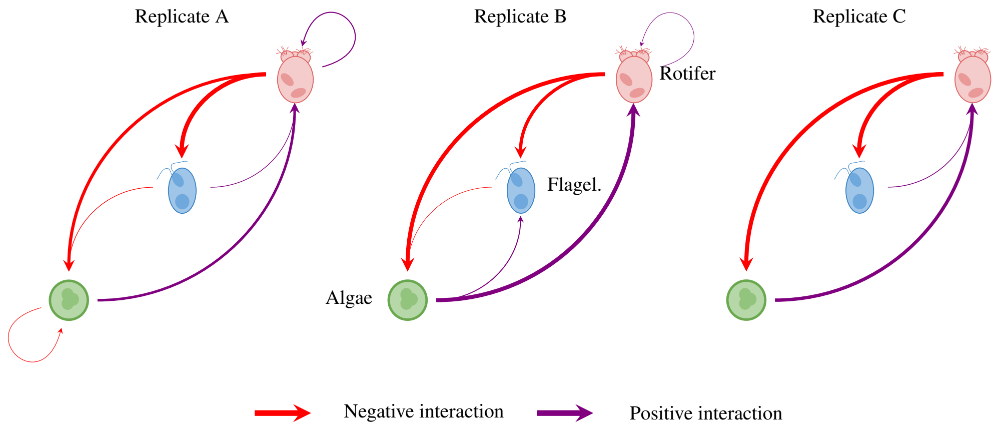

Figure 5: Interaction networks inferred from 3 replicated time series of algae, flagellate, and rotifers. This figure shows the direction and strength of ecological interactions inferred from 3 replicated sets of time series of algae, flagellate, and rotifer, using NODEs fitted by gradient matching. The replicates B and C were analysed in the same way as replicate A (see fig. 5 for details). Red and purple arrows correspond to negative or positive mean effects. We estimated mean effects by averaging effects (i.e. derivative of NODE approximated per-capita growth rates with respect to each population density) across the time series. The width of the arrows is proportional to the relative strength of the ecological interaction. We compute the relative strength as the % of total contributions attributable to either algae, flagellate, or rotifer, obtained from summing the square of contributions of each species throughout the time series. For instance in replicate A, the relative strength of the effect of rotifer on algae is found by summing the square of the red line in fig. 5 f., and computing the % of total contributions that it accounts for. We provide the value of the mean effects and relative strengths in table 2. The replicated time series were obtained by digitising the time series in Hiltunen et al. (2013).

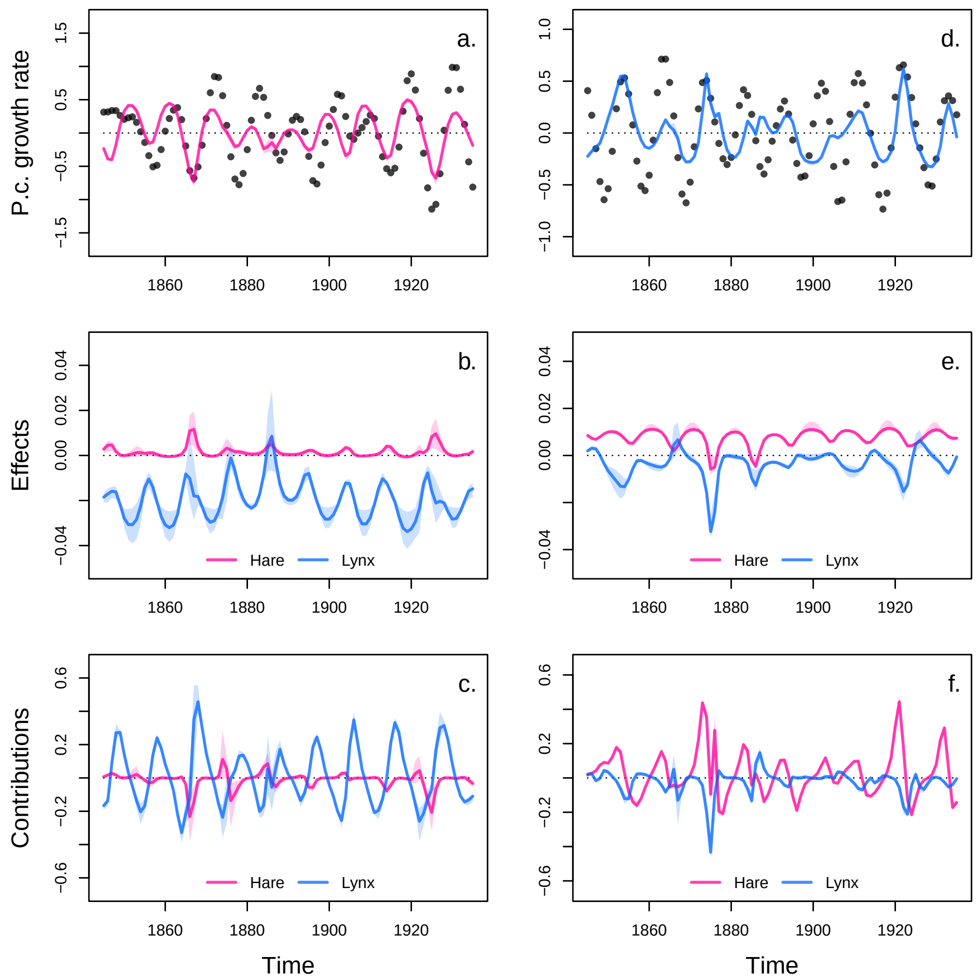

Figure 6: Drivers of dynamics of hare and lynx in the Odum and Barrett pelt count time series. This figure displays the NODE non-parametric approximations of the per-capita growth rate of hare (a., b., c.), and lynx (d., e., f.). We obtain the NODE approximations (a., d., solid line) by fitting the interpolated per-capita growth rates (black dots) with ANNs that take population densities as input. We then estimate the direction of ecological interactions (effects, b., e.) by computing the derivative of the NODE approximations with respect to each density. Finally, we compute the strength of ecological interactions (contributions, c., f.) by multiplying the interpolated dynamics of each population with its effects. The shaded area shows the 90% confidence interval, obtained by approximately sampling the posterior distributions.

Table 1: Summary of model runtimes. We measured the time required to perform 100 interpolations and 30 NODE fits to each variable in the systems. Replicate A, B, and C correspond to each replicated time series of the aglae, flagellate, and rotifer tri-trophic system (Hiltunen et al. 2013). The Hare-Lynx system correspond to the 90 years long time series of hare and lynx pelt counts (Odum and Barrett 1972). The number of time steps (N steps) is given for each time series. The total time per fit is obtain by dividing the total time in seconds by the number of fits (i.e. 130). It takes on average 5.35 minutes for the 130 NODE fits NODE, which amounts to 5.37 seconds per sample taken. This is 335 times faster than the 30 minutes fitting times obtained in a previous study (Bonnaffé, Sheldon, and Coulson 2021). These results were obtained on a macbook pro M1 MAX 2022, in base R, with non-optimised code.

			Interpolation		NODE fit			
System	N var.	N steps	N fits	time (s)	N fits	time (s)	total	total p. fit
Replicate A	3	66	100	239.47	30	129.41	368.88	6.71
Replicate B	3	66	100	233.59	30	133.13	366.72	6.77
Replicate C	3	40	100	136.51	30	74.01	210.52	3.83
Hare-lynx	2	90	100	303.64	30	33.56	337.20	4.16

Table 2: Comparison of the direction and strentgh of ecological interactions estimated by BNGM across 3 replicated tri-trophic microcosms. Mean effects are obtained by averaging the effect of one species on the growth rate of another throughout the time series. The % of total contributions is obtained by summing the square of contributions of one species density to the growth of the other at each time step throughout the time series, then by computing the proportion of total change that it accounts for. The variables G, B, and R correspond to the population density of algae, flagellate, and rotifer respectively. r^2 corresponds to the r squared of the NODE non-parametric approximation of the pre-capita growth rate of the three species.

		G	B	R
Replicate A	r^2	0.3	0.47	0.94
Mean effects	on G	-0.61	-0.85	-1.41
	on B	0.00	0.08	-0.90
	on R	2.84	0.93	1.23
% of total contributions	to G	0.13	0.15	0.73
	to G	0.00	0.00	1.00
	to R	0.60	0.16	0.25
Replicate B	r^2	0.65	0.85	0.47
Mean effects	on G	0.00	-0.56	-1.13
	on B	0.34	0.00	-0.58
	on R	0.87	0.00	0.19
% of total contributions	to G	0.00	0.06	0.94
	to B	0.23	0.00	0.77
	to R	0.95	0.00	0.05
Replicate C	r^2	0.93	0.29	0.87
Mean effects	on G	-0.14	0.13	-2.31
	on B	-0.05	-0.09	-0.72
	on R	2.46	0.49	-0.09
% of total contributions	to G	0.02	0.02	0.96
	to B	0.00	0.01	0.99
	to R	0.79	0.18	0.03

8 Supplementary

A Bayesian regularisation