

RESEARCH ARTICLE

Explaining empirical dynamic modelling using verbal, graphical and mathematical approaches

Andrew M. Edwards^{1,2}  | Luke A. Rogers¹  | Carrie A. Holt¹ 

¹Pacific Biological Station, Fisheries and Oceans Canada, Nanaimo, British Columbia, Canada

²Department of Biology, University of Victoria, Victoria, British Columbia, Canada

Correspondence

Andrew M. Edwards, Pacific Biological Station, Fisheries and Oceans Canada, 3190 Hammond Bay Road, Nanaimo, British Columbia V9T 6N7, Canada.
Email: andrew.edwards@dfo-mpo.gc.ca

Funding information

Fisheries and Oceans Canada

Abstract

Empirical dynamic modelling (EDM) is becoming an increasingly popular method for understanding the dynamics of ecosystems. It has been applied to laboratory, terrestrial, freshwater and marine systems, used to forecast natural populations and has addressed fundamental ecological questions. Despite its increasing use, we have not found full explanations of EDM in the ecological literature, limiting understanding and reproducibility. Here we expand upon existing work by providing a detailed introduction to EDM. We use three progressively more complex approaches. A short verbal explanation of EDM is then explicitly demonstrated by graphically working through a simple example. We then introduce a full mathematical description of the steps involved. Conceptually, EDM translates a time series of data into a path through a multi-dimensional space, whose axes are lagged values of the time series. A time step is chosen from which to make a prediction. The state of the system at that time step corresponds to a 'focal point' in the multi-dimensional space. The set (called the library) of candidate nearest neighbours to the focal point is constructed, to determine the nearest neighbours that are then used to make the prediction. Our mathematical explanation explicitly documents which points in the multi-dimensional space should not be considered as focal points. We suggest a new option for excluding points from the library that may be useful for short-term time series that are often found in ecology. We focus on the core simplex and S-map algorithms of EDM. Our new R package, pbsEDM, enhances understanding (by outputting intermediate calculations), reproduces our results and can be applied to new data. Our work improves the clarity of the inner workings of EDM, a prerequisite for EDM to reach its full potential in ecology and have wide uptake in the provision of advice to managers of natural resources.

KEYWORDS

attractor reconstruction, delay embedding, model-free forecasting, simplex projection, Takens' theorem

TAXONOMY CLASSIFICATION

Applied ecology, Ecosystem ecology, Population ecology

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2024 His Majesty the King in Right of Canada. *Ecology and Evolution* published by John Wiley & Sons Ltd. Reproduced with the permission of the Minister of Fisheries and Oceans Canada.

1 | INTRODUCTION

Population forecasts are widely used to provide advice to natural resource managers and to understand how ecosystems might respond to a changing environment. They are usually made by prescribing a mathematical model to describe the system, fitting the model to data to estimate parameters and iterating the model forward in time, possibly under a range of future scenarios. For single populations, such models include Ricker models (e.g. Dorner et al., 2009), matrix population models (e.g. Caswell et al., 1999) and age-structured models that require complex likelihood functions and computational Bayesian techniques to fit to data (e.g. Cleary et al., 2019). Multi-trophic models of the marine ecosystem can consist of several coupled differential equations (e.g. Fasham et al., 1990).

Ecologists do have basic equations that might be considered as laws, such as a population will exponentially grow (or decline) under a constant environment (Turchin, 2001). However, even simple single-species models consisting of a single difference equation can display qualitatively different behaviour with just slight changes in parameters (May, 1976). Multi-trophic models of the marine ecosystem also exhibit such behaviour, yet it is hard to prescribe specific values to many parameters, such as growth and mortality (Edwards, 2001; Yool, 1998). Even slight structural changes in model formulation can also drastically change predictions (Wood & Thomas, 1999).

Empirical dynamic modelling (EDM) aims to avoid such issues by not requiring a mathematical model of the system being considered (Chang et al., 2017; Deyle et al., 2013; Munch et al., 2020; Sugihara & May, 1990; Ye et al., 2015). Therefore, there is no need to estimate parameters. Rather, EDM uses only the available data to calculate forecasts. It does this by translating time series of data into a path through multi-dimensional space and making forecasts based on nearest spatial neighbours.

Recent examples demonstrate EDM's continuing application to fundamental problems in ecology. Ushio (2022) proposed a new hypothesis of how patterns of community diversity emerge, based on EDM analysis of environmental DNA data from experimental rice plots in Japan. Deyle et al. (2022) improved the quantitative understanding of food-web and chemical changes in Lake Geneva by using a novel hybrid combination of a parametric physical model and an EDM analysis of the biogeochemical variables. Rogers et al. (2022) demonstrated that chaos is not rare in natural ecosystems by analysing a global database of single-species population time series. Grziwotz et al. (2018) revealed complex environmental drivers of nine mosquito sub-populations in French Polynesia. Karakoç et al. (2020) investigated community assembly and stability through an EDM analysis of populations of laboratory microbial communities. They found changes in species interactions that were driven by the presence of a predator, behaviour that is difficult to model with traditional Lotka-Volterra-type ecological models.

Such dependence of dynamics on another variable (predators in this case) is particularly amenable to analysis using EDM because EDM does not require explicitly prescribing equations, from the

many choices available, to model such processes (Ye et al., 2015). As such, there may be a role for EDM in Ecosystem Based Fisheries Management, a current focus of several government agencies that takes into account the ecosystem when managing fisheries (Howell et al., 2021). Applications to fish populations have already been widespread, including cod (Sguotti et al., 2020), salmon (Ye et al., 2015) and tuna (Harford et al., 2017), plus forage fish such as menhaden, sardine and anchovy (Deyle et al., 2018; Sugihara et al., 2012). In simulation studies, EDM was found to provide low errors in forecasted fish recruitment (Van Beveren et al., 2021).

Here we expand upon the current literature that describes EDM, e.g. Chang et al. (2017), Deyle et al. (2013), Munch et al. (2020), Sugihara and May (1990), Ye et al. (2015), the third of which includes a useful glossary; see Munch et al. (2022) for an overview of recent advances in EDM, including methods for dealing with missing data. We build up a comprehensive description of the core simple algorithm of EDM that explicitly gives the steps involved; such steps have probably not previously been described in such detail (G. Sugihara, Scripps Institution of Oceanography, pers. comm.). We use three progressively more detailed approaches, starting with a short verbal explanation. Next we give a graphical explanation (without all the precise details) for a simple example time series. Then we extend existing notation and derive the mathematics for the univariate situation, building on the explanation by Deyle et al. (2013). Technical mathematical concepts are kept to a minimum. We extend our derivation to the multivariate situation and the S-map algorithm in Appendix S1.

Our motivation to obtain a deeper understanding of EDM originated in our desire to investigate the potential of using EDM to provide advice to fisheries managers, particularly in the context of considering ecosystem effects. We wanted to fully understand the methods so that we could write our own R package, `pbsEDM` (Rogers & Edwards, 2023), tailored to our specific applications. Despite the widespread use of EDM, we did not find a full description that explained all the steps unambiguously in sufficient detail for us to write our own independent code.

This lack of explanatory detail led to us developing the descriptions presented here to help users, particularly new ones, understand the inner workings of EDM. These descriptions include two aspects of EDM that we had not seen previously reported (though some practitioners may well be aware of them). We explicitly define the allowable focal points from which predictions can be made (aspect 1), and calculate the library (or set) of candidate nearest neighbours to use for predictions (aspect 2). This allows for a clearer understanding of how the size of the library depends on both the number of lags being considered in an analysis and on the time step from which a prediction is being made.

Note that EDM has been called 'an equation-free approach' (Ye et al., 2015) due to it not specifying equations that represent a mathematical model to represent the system. The equations we introduce here do not represent a model, but explicitly and unambiguously explain the inner workings of EDM. The mathematical details themselves are not overly technical, mainly dealing with

careful definitions of vectors and matrices, which first requires thoughtful consideration of notation, as is often the case (Edwards & Auger-Méthé, 2019).

Practitioners of EDM should be aware that it has a long and strong theoretical background (Kantz & Schreiber, 2004; Packard et al., 1980; Schaffer & Kot, 1986; Stark et al., 2003; Takens, 1981), but do not need to understand the full details.

In our examples, the variables represent population numbers and associated environmental variables, but the methods are applicable to time series of any quantities in ecology or other fields. This does require that the system is not completely stochastic, and has some underlying deterministic rules (that may still be subject to some randomness). Our intention is for our `pbsEDM` R package to complement the popular R package `rEDM` (Park et al., 2023, with a tutorial at <https://github.com/SugiharaLab/rEDM/blob/master/vignettes/rEDM-tutorial.pdf>) and Python package `pyEDM` (Park & Smith, 2023), to aid understanding and reproducibility. All intermediate calculations are available as output in `pbsEDM` and all code is in R, while `rEDM` contains C++ code (which is faster than R code but less readable than R to many ecologists); however, `rEDM` and `pyEDM` also include advanced algorithms that are not in `pbsEDM`. All code for reproducing our calculations and figures (each as a single function), and for applying methods to users' own data, is publicly available within `pbsEDM` (and File S1).

2 | EXPLAINING EDM VERBALLY

The idea behind EDM is that we start with a simple time series of a variable, such as the annual values of the size of a population. We then construct additional time series of first-differenced values (differences between consecutive values) and lags of those values (which compare the first-differenced values with those in the past, such as 1 year previously or 2 years previously). These first-differenced and lagged values then make up the components of vectors that can be plotted as points in a multi-dimensional space known as a *state space*. Joining these points together in temporal order traces a path through the state space. To make a prediction from a point in the state space, the simplex algorithm finds the nearest neighbours (in terms of spatial distance in the state space) and sees where those neighbours went in their subsequent time step. A weighted average of these destinations yields the prediction. The key concept of EDM is the transforming of the time series of single values at each point in time, into points that lie in the multi-dimensional state space. The points in the state space can reveal a geometric structure that is not apparent when viewing the data as a simple time series.

The reason that EDM can work is because the lagged values contain intrinsic information concerning the system. Ye et al. (2015) described the general concept as “local neighborhoods (and their trajectories) in the reconstruction [our state space] map to local neighborhoods (and their trajectories) of the original system”.

To expand on our brief verbal description, we now work through an analysis of an example time series using graphical explanations

and then derive an explicit mathematical description of the simplex algorithm.

3 | EXPLAINING EDM GRAPHICALLY

3.1 | Plotting values from a simple time series in various ways

We start with a simple example time series of simulated data generated from a stochastic salmon population model that gives populations at times $t = 1, 2, 3, \dots, 100$. The time series is used for illustrative purposes, not to make any claims about applying EDM to any particular scenario. Figure 1 shows how the data can be plotted in five different ways, to introduce the concepts underpinning EDM.

Figure 1a shows the simple time series of the size of the population at time t , N_t , with generally low and occasionally high values. In Figure 1b the same data are shown with N_t plotted against the value at the previous time step, namely N_{t-1} . This is a phase plane because there are two axes that represent variables and there is no time axis. Figure 1c introduces the first-differenced values, defined as the difference between the population at the next time step and the current time step, that is, $Y_t = N_{t+1} - N_t$. By definition, Y_t can take negative and positive values (while $N_t \geq 0$ since it represents the size of a population).

Figure 1d shows the phase plane of each Y_t against its lagged value Y_{t-1} , revealing some geometric structure in the data. This structure is inherently coming from the population N_t being mostly at low levels, but with occasional high values followed by immediate drops back down to low levels. There is clearly a cluster of points around the origin, representing low values of Y_t (and Y_{t-1}) for most of the time series, due to small changes between consecutive values of N_t . There are three ‘arms’ along which the remaining points lie, plus empty areas of the phase plane that never appear to be visited. The central top arm contains points representing values of Y_{t-1} close to zero that are then followed by a large value of Y_t (a large increase in the population). When $t = 98$ the system is at the location $(Y_{t-1}, Y_t) = (Y_{97}, Y_{98})$, as indicated by the red open circle in Figure 1d. At the next time step, $t = 99$, the system moves to (Y_{98}, Y_{99}) , given by the red closed circle. A graphical way to view this is shown by the red lines in Figure 1d. First, trace horizontally from (Y_{97}, Y_{98}) to the 1:1 line $Y_{t-1} = Y_t$, so that the previous Y_t value on the y-axis (Y_{98}) becomes the new Y_{t-1} value on the x-axis (when t increases by 1, the old Y_t becomes the new Y_{t-1}). Then trace up or down to reach the new Y_t value. This approach, inspired by the dynamical systems concept of ‘cobwebbing’ (Murray, 1989), leads to the tracing out of the path of the grey lines in Figure 1d. This path is always clockwise. For the full time series, this results in the bottom-right arm, for which a large value of Y_{t-1} is always immediately followed by a large negative Y_t (a large decline). Continuing clockwise leads to the left arm, for which the large declines are followed by very minor changes close to zero, and so in the next time step the trajectory heads back into the central cluster.

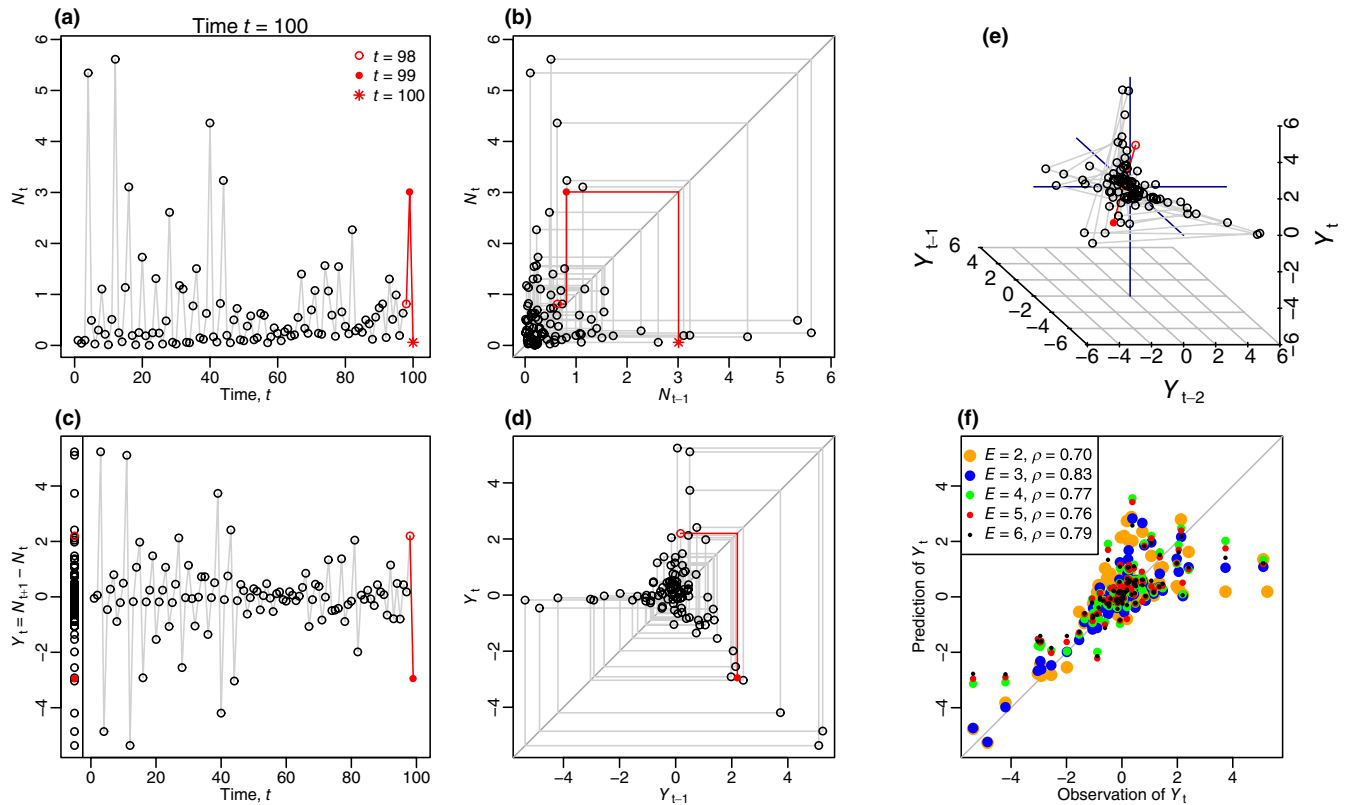


FIGURE 1 Different ways of plotting a simple simulated time series. Panels are arranged so that the y axes are the same for (a) and (b), and then for (c), (d) and (f). (a) Population values N_t (units of 100,000 individuals) are shown through time. The final time step ($t = 100$) is shown in the title, and the final three values of N_t are shown in red using the symbols indicated (these carry through to the other panels except (f)). (b) Same values in a phase plane, with N_t against N_{t-1} . The grey lines demonstrate the points progressing clockwise around the phase plane (see text). (c) Time series of resulting first-differenced values $Y_t = N_{t+1} - N_t$, with all values also overlaid in a single column to the left of $t = 0$. (d) Phase plane of Y_t against Y_{t-1} reveals a geometric structure that is not apparent in the preceding panels. (e) Extends (d) to three dimensions, showing Y_t , Y_{t-1} and Y_{t-2} . (f) The predicted results and corresponding Pearson correlation coefficients, ρ , of the observed values of Y_t for different values of embedding dimension E . [Figure A.1](#): Appendix S1 shows a controllable frame-by-frame animation of this figure for $t = 1$ to $t = 100$, and [Movie S1](#) gives a narrated version.

The cobwebbing idea graphically shows that, for example, when the population experiences a large increase (large Y_t), the next few time steps are expected to follow a certain path clockwise around the phase plane (namely, a large decrease in Y_t shown in the bottom-right arm, followed by a small value of Y_t close to zero in the left arm). The idea of EDM is to harness such geometric structure in the spatial phase plane to make predictions in the time dimension.

[Figure 1d](#) also shows empty regions that the system does not visit, namely the top-right area (a large increase of N_t is never followed by another large increase: Y_{t-1} and Y_t are never both large), the bottom-left area (a decline or slight increase is never followed by a large decline: Y_{t-1} and Y_t are never both very negative), and the top-left area (a large decline is never followed by a large increase: a very negative Y_{t-1} is never followed by a large Y_t). This last description translates to N_t never going high, then low, then immediately high again.

The structure in [Figure 1d](#) represents the attractor on which the system evolves through time. This gives a useful way of thinking about EDM – the fundamental description of the dynamics of the system can be thought of as being given by the observed attractor

(based solely on the data), rather than by a prescribed set of equations (Munch et al., 2020).

[Figure 1e](#) extends the two-dimensional phase plane idea to three dimensions, showing Y_t against Y_{t-1} and Y_{t-2} . While [Figure 1d](#) included a lag of one time step, [Figure 1e](#) includes lags of one (Y_{t-1}) and two (Y_{t-2}) time steps. Again, this reveals an underlying geometric structure of the system (seen more clearly in the animated [Figure A.1](#): Appendix S1 which shows the structure being built up through time). The three dimensions correspond, in EDM language, to an embedding dimension of $E = 3$, because the points are embedded in three-dimensional space. This space is known as the *state space* (the multi-dimensional equivalent of the two-dimensional phase plane). The phase plane in [Figure 1d](#) corresponds to $E = 2$. Higher embedding dimensions (4, 5, 6, etc.) are also used, but obviously not easily plotted. The points at the left of [Figure 1c](#) show the distribution of values of Y_t in one dimension, which is essentially an embedding dimension of $E = 1$. This is not commonly used in EDM but is shown here to illustrate how we can have the points on a line for $E = 1$, on a phase plane ([Figure 1d](#)) for $E = 2$ and a three-dimensional plot ([Figure 1e](#)) for $E = 3$.

In hindsight, some of the aforementioned conclusions from Figure 1d can be teased out from Figures 1a,c, but the phase plane in (d) makes them much more apparent. However, EDM utilises structure in higher dimensions (i.e. using more lags) that cannot be easily visualised and cannot be inferred from the simple time series. Certainly, the structure in the three-dimensional Figure 1e cannot be easily ascertained from the simple time series.

Figures 1a–e have simply plotted the data in different ways, there have been no statistical analyses or calculations beyond first-differencing and lagging. Such plotting has revealed some structure behind the time series that is not immediately apparent in the simple time series plots. Figure 1f is discussed after we explain how EDM uses the geometric structure to make predictions.

3.2 | Graphically demonstrating the simplex algorithm

For our example time series, we first choose a focal time t^* , which means that we want to use EDM to predict where the system goes in the subsequent time step (Deyle et al., 2013). We choose, as an example, $t^* = 39$, such that we want to estimate Y_{40} (where the system goes in the next time step) given knowledge of the rest of the time series. We denote the estimated value as \hat{Y}_{40} , and more generally, for a given t^* we want to estimate \hat{Y}_{t^*+1} .

We can then compare the predicted value \hat{Y}_{40} to its known value to see how well the simplex algorithm performs for $t^* = 39$. The state of the system at $t^* = 39$ is highlighted in Figure 2, showing the values of $Y_{t-1} = Y_{38}$ and $Y_t = Y_{39}$ in the lagged phase plane. For the phase plane the nearest three neighbours are located (red circles). These are the nearest neighbours spatially, but this does not mean that they are close to each other in time; the actual times of these points are $t = 11, 43$, and 98 . The crux of EDM is to see where these points move to in the phase plane in their next time step, to make a prediction of where the focal point will go. The idea being that close points in the phase plane will move to close points for their subsequent time step, and this structure in the system allows us to estimate $\hat{Y}_{t^*+1} = \hat{Y}_{40}$.

The purple arrows in Figure 2 show where the three nearest neighbours move to in their subsequent time steps $t = 12, 44$, and 99 (recall from the cobwebbing idea that the Y_t values become the new Y_{t-1} values, and so it is only the new Y_t values that give new information). A weighted average of these new Y_t values then gives our prediction of $\hat{Y}_{t^*+1} = \hat{Y}_{40}$ (the blue star), which here is close to the value of Y_{40} already known from our time series (green circle). The weighting is based on the relative closeness of the three nearest neighbours to the focal point (explained in detail later).

We can make similar predictions for all alternative values of the focal time t^* (in addition to $t^* = 39$), and evaluate how the predicted values \hat{Y}_{t^*+1} compare to the known values Y_t . We then calculate the Pearson correlation coefficient (ρ) of these, which is the usual, but not the only, way to characterise the performance of EDM predictions (Ye et al., 2015), with $\rho = 1.0$ representing a perfect positive

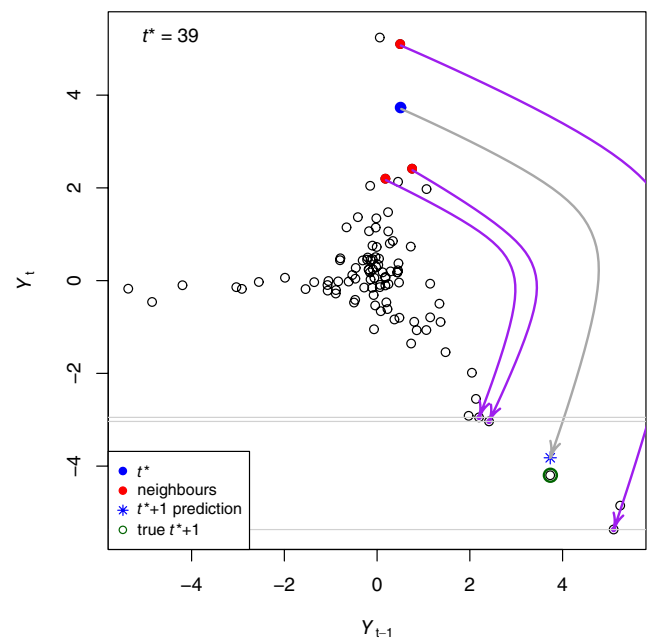


FIGURE 2 For the example time series, embedding dimension $E = 2$ yields the phase plane of Y_t against Y_{t-1} as in Figure 1d. Given focal time $t^* = 39$ (blue circle indicating Y_{38} and Y_{39}) we want to predict $Y_{t^*+1} = Y_{40}$. The three nearest neighbours to the blue circle in the phase plane are shown by the red circles. These are at times 11, 43 and 98, though the times cannot be inferred from the phase plane. The purple arrows show where these points move to in the phase plane one time step later, namely to the points corresponding to times 12, 44 and 99. A weighted average of Y_{12} , Y_{44} and Y_{99} (grey horizontal lines) then gives the estimate of Y_{40} (blue star). In this case it is close to the known true value of Y_{40} (green circle). An annotated animation of this figure is shown in Figure A.2: Appendix S1, and the figures for all valid t^* values are shown in Figure A.3: Appendix S1.

correlation between observations and predictions. For the phase plane from Figure 2, which has embedding dimension $E = 2$, we have $\rho = 0.70$.

This idea is then repeated for prescribed embedding dimensions of $E = 3, 4, 5, \dots$, and the predicted and observed values up to $E = 6$ are shown in Figure 1f, together with the corresponding ρ . The best-performing (highest ρ) embedding dimension is $E = 3$ (Figure 3), and this is the dimension that would, therefore, be used in EDM to forecast the population into the future, beyond the timespan of the data.

4 | EXPLAINING EDM MATHEMATICALLY

We now develop the above ideas using a more formal mathematical approach. We mostly follow and adapt the description given by Deyle et al. (2013), extending their work to give a full mathematical description that leads to a deeper understanding of EDM and its limitations. The simplex algorithm is described for a univariate time series, such as an annual survey estimate of a population, and extended to the multivariate case and the S-map technique (Sugihara, 1994)

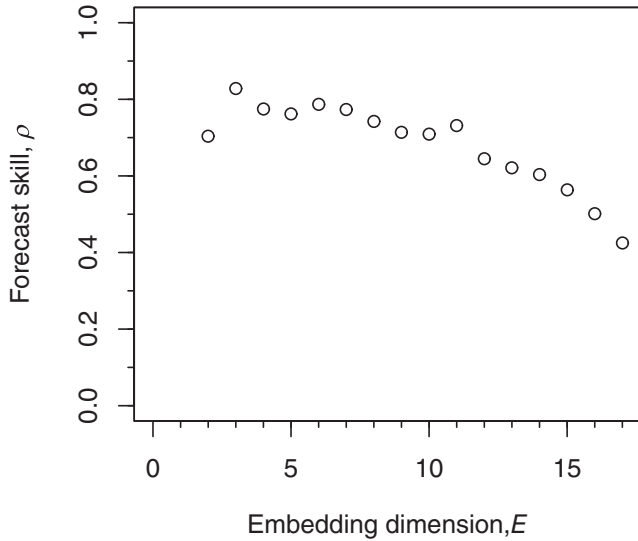


FIGURE 3 Dependence of the Pearson correlation coefficient, ρ , on embedding dimension, E , for the example time series from Figure 1. The best fitting model is given by the highest ρ , and corresponds to $E = 3$ which is what would be used to forecast \hat{N}_{T+1} . We have shown high enough values of E to show a clear decline in ρ , but in general the maximum E considered should be about \sqrt{T} (Munch et al., 2020), which is 10 here.

in Appendix S1. Notation is extended from that defined clearly by Deyle et al. (2013), and summarised in Table 1 for reference.

4.1 | Algorithm for simplex projection

We consider a univariate time series of population size N_t at each time $t = 1, 2, 3, \dots, T$. As earlier, we first-difference the data to give scalars

$$Y_t = N_{t+1} - N_t \quad (1)$$

for $t = 1, \dots, T - 1$, such that the first value, Y_1 , is defined, with Y_T undefined. First-differencing is often done to help remove any simple linear mean trend (Chang et al., 2017). The aim of the analysis is to estimate \hat{N}_{T+1} , that is, the population the year after the final year of data, by estimating \hat{Y}_T and then rearranging (1) to give $\hat{N}_{T+1} = \hat{Y}_T + N_T$.

The simplex algorithm was detailed as steps (i) to (vii) by Deyle et al. (2013). These are summarised and extended in Table 2 to give an overall idea of the approach and then expanded upon here.

(i) For a given embedding dimension E , we define the vector $\tilde{\mathbf{x}}_t$ in lagged space as containing Y_t and consecutive lags down to Y_{t-E+1} :

$$\tilde{\mathbf{x}}_t = \begin{bmatrix} Y_t \\ Y_{t-1} \\ \dots \\ Y_{t-E+2} \\ Y_{t-E+1} \end{bmatrix}. \quad (2)$$

TABLE 1 The main notation used here.

Notation	Definition
<i>Indices</i>	
t	Index for time; $t = 1, 2, 3, \dots, T$
T	Number of time steps
t^*	Focal time at which we know the state of the system and want to predict the state at $t^* + 1$
<i>Variables</i>	
N_t	Value, such as population size, at $t = 1, \dots, T$
Y_t	First-difference value $Y_t = N_{t+1} - N_t$
$\hat{\cdot}$	Estimate of \cdot
<i>EDM calculations</i>	
$\tilde{\mathbf{x}}_t$	Vector of length E defining the axes of the lagged state space, for example, $\tilde{\mathbf{x}}_t = (Y_t, Y_{t-1}, Y_{t-2})$
\mathbf{x}_t	Realised values of the components of $\tilde{\mathbf{x}}_t$, for example, $\mathbf{x}_t = (3, -5, 1)$; each element of \mathbf{x}_t is the value along each axis of the E -dimensional space, where the axes are defined by components of $\tilde{\mathbf{x}}_t$
\mathbf{x}_{t^*}	Realised values of the components of $\tilde{\mathbf{x}}_t$ at the focal time $t = t^*$
E	Embedding dimension, the number of dimensions of the state space in which the system is being embedded to look for the nearest $E + 1$ neighbours to the focal point \mathbf{x}_{t^*} ; E is the length of $\tilde{\mathbf{x}}_t$
\mathbf{X}	Matrix with rows representing time and columns representing each of the E components of $\tilde{\mathbf{x}}_t$; row t represents the system state at time t with the j th element representing the j th component of \mathbf{x}_t
$\mathcal{L}(E, t^*)$	Library for a given E and t^* , consisting of the set of \mathbf{x}_t that are candidates to be considered as nearest neighbours of \mathbf{x}_{t^*} .
C_{E, t^*}	The number of vectors \mathbf{x}_t in the library $\mathcal{L}(E, t^*)$
C_E	The usual value of C_{E, t^*} for a given E , defined as $C_E = T - 2(E + 1)$; $C_{E, t^*} \geq C_E$
ψ_i	After calculating the distance between \mathbf{x}_{t^*} and each \mathbf{x}_t in the library, ψ_1 gives the time index of the \mathbf{x}_t that is the nearest neighbour to \mathbf{x}_{t^*} , ψ_2 corresponds to the second nearest neighbour, etc.

So $\tilde{\mathbf{x}}_t$ has length E with each element defining an axis that we will be using to construct the E -dimensional state space. Actual realised values (numbers) for a particular t are recorded in vectors \mathbf{x}_t , with each element referring to its corresponding axis definition in $\tilde{\mathbf{x}}_t$. For example, with our simulated time series from Figure 1, $E = 4$ yields

$$\tilde{\mathbf{x}}_t = \begin{bmatrix} Y_t \\ Y_{t-1} \\ Y_{t-2} \\ Y_{t-3} \end{bmatrix} \text{ yielding } \mathbf{x}_4 = \begin{bmatrix} Y_4 \\ Y_3 \\ Y_2 \\ Y_1 \end{bmatrix} = \begin{bmatrix} -4.854 \\ 5.241 \\ 0.059 \\ -0.057 \end{bmatrix}, \mathbf{x}_5 = \begin{bmatrix} Y_5 \\ Y_4 \\ Y_3 \\ Y_2 \end{bmatrix} = \begin{bmatrix} -0.461 \\ -4.854 \\ 5.241 \\ 0.059 \end{bmatrix}, \dots, \quad (3)$$

with $\tilde{\mathbf{x}}_t$ defining the axes of the state space.

The components of $\tilde{\mathbf{x}}_t$ from (3) give the column headings of matrix \mathbf{X} :

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_{T-2} \\ \mathbf{x}_{T-1} \\ \mathbf{x}_T \end{bmatrix} = \begin{bmatrix} Y_t & Y_{t-1} & Y_{t-2} & Y_{t-3} \\ Y_1 & \times & \times & \times \\ Y_2 & Y_1 & \times & \times \\ Y_3 & Y_2 & Y_1 & \times \\ Y_4 & Y_3 & Y_2 & Y_1 \\ Y_5 & Y_4 & Y_3 & Y_2 \\ \vdots & \vdots & \vdots & \vdots \\ Y_{T-2} & Y_{T-3} & Y_{T-4} & Y_{T-5} \\ Y_{T-1} & Y_{T-2} & Y_{T-3} & Y_{T-4} \\ \times & Y_{T-1} & Y_{T-2} & Y_{T-3} \end{bmatrix}, \quad (4)$$

for which row t consists of the explicit values of \mathbf{x}_t (implicitly understood to be written here as a row vector) with undefined values indicated by \times . Vectors $\mathbf{x}_1, \mathbf{x}_2$ and \mathbf{x}_3 are undefined because the Y_t for $t \leq 0$ are undefined. Some brief descriptions of the simplex algorithm do not mention that some points should be excluded from the library (e.g. Hsieh et al., 2005; Ye et al., 2015), while Deyle et al. (2013) did note that the first few time values will not have a vector in the state

TABLE 2 The steps of the simplex algorithm (extended from Deyle et al., 2013).

Step	Brief description
(i)	Translate the time series values into vectors in the multi-dimensional state space defined by a given embedding dimension E
(ii)	Pick a focal time from which to predict
(iii)	Define the set of library vectors of candidate nearest neighbours to the focal point
(iv)	Calculate the distances between appropriate points in the state space
(v)	Identify the nearest neighbours to the focal point
(vi)	Make a prediction using a weighted average of the known next positions of the nearest neighbours
(vii)	Repeat steps (ii)–(vi) for all appropriate focal times
(viii)	Calculate the correlation coefficient between predictions and the known observations
(ix)	Repeat steps (i)–(viii) for different values of E , using the optimal one (E with maximum correlation coefficient) to forecast the future value of the population

TABLE 3 Two aspects of EDM that we document here to aid new users of EDM.

Aspect	Brief description
1	The allowable focal point times t^* (from which to make predictions) depend explicitly upon the embedding dimension E . They require lagged values that do not extend before the start of the time series or beyond the end of it
2	We explicitly calculate the library of candidate nearest neighbours of the focal point, and derive a new relationship showing how the size of the library depends on both t^* and E

space; here we make that more explicit. Also, \mathbf{x}_T does not exist because Y_T is undefined in (1); however, we include it in \mathbf{X} because we will want to forecast Y_T and it is helpful for \mathbf{X} to have T rows.

Matrix (4) is for $E = 4$. Extending this for a general value of E we have

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_{E-1} \\ \mathbf{x}_E \\ \mathbf{x}_{E+1} \\ \vdots \\ \mathbf{x}_{T-4} \\ \mathbf{x}_{T-3} \\ \mathbf{x}_{T-2} \\ \mathbf{x}_{T-1} \\ \mathbf{x}_T \end{bmatrix} = \begin{bmatrix} Y_t & Y_{t-1} & Y_{t-2} & Y_{t-3} & \cdots & Y_{t-E+2} & Y_{t-E+1} \\ Y_1 & \times & \times & \times & \cdots & \times & \times \\ Y_2 & Y_1 & \times & \times & \cdots & \times & \times \\ Y_3 & Y_2 & Y_1 & \times & \cdots & \times & \times \\ Y_4 & Y_3 & Y_2 & Y_1 & \cdots & \times & \times \\ Y_5 & Y_4 & Y_3 & Y_2 & \cdots & \times & \times \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_{E-1} & Y_{E-2} & Y_{E-3} & Y_{E-4} & \cdots & Y_1 & \times \\ Y_E & Y_{E-1} & Y_{E-2} & Y_{E-3} & \cdots & Y_2 & Y_1 \\ Y_{E+1} & Y_E & Y_{E-1} & Y_{E-2} & \cdots & Y_3 & Y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_{T-4} & Y_{T-5} & Y_{T-6} & Y_{T-7} & \cdots & Y_{T-E-2} & Y_{T-E-3} \\ Y_{T-3} & Y_{T-4} & Y_{T-5} & Y_{T-6} & \cdots & Y_{T-E-1} & Y_{T-E-2} \\ Y_{T-2} & Y_{T-3} & Y_{T-4} & Y_{T-5} & \cdots & Y_{T-E} & Y_{T-E-1} \\ Y_{T-1} & Y_{T-2} & Y_{T-3} & Y_{T-4} & \cdots & Y_{T-E+1} & Y_{T-E} \\ \times & Y_{T-1} & Y_{T-2} & Y_{T-3} & \cdots & Y_{T-E+2} & Y_{T-E+1} \end{bmatrix}. \quad (5)$$

For a given time series of length T , the larger the value of E , the larger the size of the upper-right triangle of undefined values, because a larger embedding dimension requires more lagged values. The first row that is fully known (requiring that Y_{t-E+1} exists) is when $t = E$.

(ii) Pick a focal time t^* for which we know Y_{t^*} and want to predict the value of Y_{t^*+1} , with the prediction denoted \hat{Y}_{t^*+1} . In the E -dimensional state space, we do this by requiring knowledge of the full \mathbf{x}_{t^*} and then estimating $\hat{\mathbf{x}}_{t^*+1}$ to give us our estimate of \hat{Y}_{t^*+1} from (2). Not all values of t are available to use for t^* (aspect 1; briefly explained in Table 3), which will be made explicit shortly. Note that we call \hat{Y}_{t^*+1} for general t^* a 'prediction', reserving the term 'forecast' for estimating future \hat{Y}_T and \hat{N}_{T+1} beyond the existing data.

(iii) Given t^* , define the library $\{\mathbf{x}_t\}$ of candidate nearest neighbours of \mathbf{x}_{t^*} . To determine the library $\{\mathbf{x}_t\}$ we start with an expanded version of \mathbf{X} from (5) for general t^* , and systematically cross out Y_{t^*+1} and various \mathbf{x}_t that must be excluded from the library due to four conditions, resulting in

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \\ \vdots \\ \mathbf{x}_{E-1} \\ \mathbf{x}_E \\ \mathbf{x}_{E+1} \\ \vdots \\ \mathbf{x}_{t^*-1} \\ \mathbf{x}_{t^*} \\ \mathbf{x}_{t^*+1} \\ \mathbf{x}_{t^*+2} \\ \vdots \\ \mathbf{x}_{t^*+E} \\ \mathbf{x}_{t^*+E+1} \\ \mathbf{x}_{t^*+E+2} \\ \vdots \\ \mathbf{x}_{T-4} \\ \mathbf{x}_{T-3} \\ \mathbf{x}_{T-2} \\ \mathbf{x}_{T-1} \\ \mathbf{x}_T \end{pmatrix} = \begin{pmatrix} Y_t & Y_{t-1} & Y_{t-2} & Y_{t-3} & \cdots & Y_{t-E+2} & Y_{t-E+1} \\ Y_1 & \times & \times & \times & \cdots & \times & \times \\ Y_2 & Y_1 & \times & \times & \cdots & \times & \times \\ Y_3 & Y_2 & Y_1 & \times & \cdots & \times & \times \\ Y_4 & Y_3 & Y_2 & Y_1 & \cdots & \times & \times \\ Y_5 & Y_4 & Y_3 & Y_2 & \cdots & \times & \times \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_{E-1} & Y_{E-2} & Y_{E-3} & Y_{E-4} & \cdots & Y_1 & \times \\ Y_E & Y_{E-1} & Y_{E-2} & Y_{E-3} & \cdots & Y_2 & Y_1 \\ Y_{E+1} & Y_E & Y_{E-1} & Y_{E-2} & \cdots & Y_3 & Y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_{t^*-1} & Y_{t^*-2} & Y_{t^*-3} & Y_{t^*-4} & \cdots & Y_{t^*-E+1} & Y_{t^*-E} \\ Y_{t^*} & Y_{t^*-1} & Y_{t^*-2} & Y_{t^*-3} & \cdots & Y_{t^*-E+2} & Y_{t^*-E+1} \\ \mathbf{x}_{t^*+1} & \mathbf{x}_{t^*+1} & Y_{t^*} & Y_{t^*-1} & \cdots & Y_{t^*-E+3} & Y_{t^*-E+2} \\ \mathbf{x}_{t^*+2} & Y_{t^*+2} & \mathbf{x}_{t^*+1} & Y_{t^*} & Y_{t^*-1} & \cdots & Y_{t^*-E+4} & Y_{t^*-E+3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{x}_{t^*+E} & Y_{t^*+E} & Y_{t^*+E-1} & Y_{t^*+E-2} & Y_{t^*+E-3} & \cdots & Y_{t^*+2} & \mathbf{x}_{t^*+1} \\ \mathbf{x}_{t^*+E+1} & Y_{t^*+E+1} & Y_{t^*+E} & Y_{t^*+E-1} & Y_{t^*+E-2} & \cdots & Y_{t^*+3} & Y_{t^*+2} \\ \mathbf{x}_{t^*+E+2} & Y_{t^*+E+2} & Y_{t^*+E+1} & Y_{t^*+E} & Y_{t^*+E-1} & \cdots & Y_{t^*+4} & Y_{t^*+3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{x}_{T-4} & Y_{T-4} & Y_{T-5} & Y_{T-6} & Y_{T-7} & \cdots & Y_{T-E-2} & Y_{T-E-3} \\ \mathbf{x}_{T-3} & Y_{T-3} & Y_{T-4} & Y_{T-5} & Y_{T-6} & \cdots & Y_{T-E-1} & Y_{T-E-2} \\ \mathbf{x}_{T-2} & Y_{T-2} & Y_{T-3} & Y_{T-4} & Y_{T-5} & \cdots & Y_{T-E} & Y_{T-E-1} \\ \mathbf{x}_{T-1} & Y_{T-1} & Y_{T-2} & Y_{T-3} & Y_{T-4} & \cdots & Y_{T-E+1} & Y_{T-E} \\ \mathbf{x}_T & \times & Y_{T-1} & Y_{T-2} & Y_{T-3} & \cdots & Y_{T-E+2} & Y_{T-E+1} \end{pmatrix} \quad (6)$$

The four conditions for excluding components from the library of candidate nearest neighbours are:

- \mathbf{x}_t cannot be a nearest neighbour to itself—excludes \mathbf{x}_t ;
- Several \mathbf{x}_t are not fully defined (contain \times)—excludes $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{E-1}$ and \mathbf{x}_T ;
- Exclude any t for which we do not know \mathbf{x}_{t+1} (since we need to know where the nearest neighbours go in the subsequent time step in (vi))—excludes \mathbf{x}_{T-1} ;
- It may not be appropriate to use any \mathbf{x}_t that includes Y_{t+1} , since we are trying to predict Y_{t+1} —excludes $\mathbf{x}_{t^*+1}, \mathbf{x}_{t^*+2}, \dots, \mathbf{x}_{t^*+E}$. This is further investigated later.

The resulting library is given by the remaining set of \mathbf{x}_t vectors that are not crossed out in (6), namely:

$$\mathcal{L}(E, t^*) = \{\mathbf{x}_E, \mathbf{x}_{E+1}, \dots, \mathbf{x}_{t^*-2}, \mathbf{x}_{t^*-1}, \mathbf{x}_{t^*+E+1}, \mathbf{x}_{t^*+E+2}, \dots, \mathbf{x}_{T-2}\}, \quad (7)$$

for those \mathbf{x}_t that are defined, where the notation $\mathcal{L}(E, t^*)$ emphasises that the library depends upon both E and t^* ; this is aspect 2. For a time series of length $T = 50$, Figure 4 shows how the size of the library, C_{E, t^*} , varies with E and t^* .

Returning to the idea from (ii) of determining the valid values of t^* , we now flesh out aspect 1, determining the allowable values of t^* and how these depend on E . Matrix \mathbf{X} in (6) shows that \mathbf{x}_t is not defined for $t = 1, 2, \dots, E - 1$, such that t^* cannot take these values; this

is the top-right grey region of Figure 4. Also, \mathbf{x}_T is not defined and so t^* cannot equal T . The value $t^* = T - 1$ is also excluded because in step (viii) we want to compare the prediction with the known state, which cannot be done for $t^* = T - 1$ since $\mathbf{x}_{t^*+1} = \mathbf{x}_T$ is not defined. Though note that $t^* = T - 1$ can later be used to forecast $\hat{\mathbf{x}}_T$ and hence \hat{Y}_T and \hat{N}_{T+1} , which is the aim of the analysis. Excluding T and $T - 1$ corresponds to the bottom grey region in Figure 4, leaving allowable values for the focal time t^* of $E, E + 1, \dots, T - 3, T - 2$, which are the non-grey combinations in Figure 4.

Usually, the library has $C_E = T - 2(E + 1)$ components, as indicated by the bulk of values for each value of E in Figure 4 (e.g. $C_2 = 44$, since $T = 50$). Intuitively, the library has fewer components as E gets larger because the larger E uses more temporal lags which creates a higher dimensional state space, resulting in \mathbf{X} having more columns and subsequently more \mathbf{x}_t crossed out in (6) and more grey area at the top (small t^* values) of Figure 4.

However, the library size also depends on t^* , and we denote it by C_{E, t^*} . We now derive C_{E, t^*} , as used to create Figure 4. For a given E , $C_{E, t^*} > C_E$ for certain values of t^* , because for larger t^* the crossed out $\mathbf{x}_{t^*+1}, \mathbf{x}_{t^*+2}, \dots, \mathbf{x}_{t^*+E}$ components in the middle rows of (6) overlap with the crossed out \mathbf{x}_{T-1} and \mathbf{x}_T components, or do not exist since they have times $> T$. This overlapping first happens when $t^* + E = T - 1$, such that $t^* = T - E - 1$ and the library is

$$\mathcal{L}(E, t^*) = \mathcal{L}(E, T - E - 1) = \{\mathbf{x}_E, \mathbf{x}_{E+1}, \dots, \mathbf{x}_{T-E-3}, \mathbf{x}_{T-E-2}\}, \quad (8)$$

which has size $C_{E, t^*} = C_{E, T-E-1} = T - E - 2 - (E - 1) = C_E + 1$. In Figure 4, this corresponds to $C_{2, 47} = 45$ (for $E = 2$) and $C_{3, 46} = 43$ (for $E = 3$).

For the next value, $t^* = T - E$, we have

$$\mathcal{L}(E, T - E) = \{\mathbf{x}_E, \mathbf{x}_{E+1}, \dots, \mathbf{x}_{T-E-2}, \mathbf{x}_{T-E-1}\}, \quad (9)$$

which has size $C_{E, t^*} = C_{E, T-E} = T - E - 1 - (E - 1) = C_E + 2$, corresponding to $C_{2, 48} = 46$ and $C_{3, 47} = 44$ in Figure 4.

This pattern incrementally increases until we get to $t^* = T - 2$, near the end of the time series, for which \mathbf{x}_{t^*+1} and \mathbf{x}_{T-1} are the same. So in (6), \mathbf{x}_{T-1} gets excluded both because we do not know \mathbf{x}_T (exclusion condition c) and because it contains Y_{t^*+1} (condition d). This overlap means that the library is

$$\mathcal{L}(E, T - 2) = \{\mathbf{x}_E, \mathbf{x}_{E+1}, \dots, \mathbf{x}_{T-4}, \mathbf{x}_{T-3}\}, \quad (10)$$

which has size $C_{E, T-2} = T - 3 - (E - 1) = C_E + E$. In Figure 4, for $E = 2$ this is the aforementioned $C_{2, 48} = 46$, and for $E = 3$, this is, $C_{3, 48} = 45$.

In summary, the library is given by (7) and is bigger for relatively large t^* , with size explicitly given by

$$C_{E, t^*} = \begin{cases} T - 2(E + 1) = C_E, & t^* = E, E + 1, \dots, T - E - 2, \\ t^* - E, & t^* = T - E - 1, T - E, \dots, T - 2. \end{cases} \quad (11)$$

(ix) Repeat steps (i) to (viii) for a sequence of embedding dimensions E . The E that gives the highest ρ is considered to perform best, namely $E = 3$ (giving $\rho = 0.83$) for our example time series (Figure 3). That E is used to forecast the future value of the population, $\hat{N}_{T+1} = \hat{N}_{101}$, by setting $t^* = 99$ to estimate \hat{Y}_{100} and rearranging (1) to give $\hat{N}_{101} = \hat{Y}_{100} + N_{100}$. Note that, regarding aspect 1, $t^* = T - 1 = 99$ is allowed here for forecasting \hat{N}_{101} ; its exclusion in (6) is only for determining ρ . If ρ increases with E such that there is no optimal E , this suggests a high-dimensional essentially random process for all practical purposes, such that the system is difficult to model (Hsieh et al., 2005).

For our simulated data and $E = 3$, our pbsEDM implementation of steps (i)–(ix) gives $\hat{Y}_{100} = -0.077$ yielding $\hat{N}_{101} = -0.077 + 0.060 = -0.017$. Thus, the forecast is of a negative population, which is obviously unrealistic. Predictions of the first-differenced \hat{Y}_t are weighted averages of observed values of Y_t , so they must lie within the range of the observed values (e.g. Figure 2). More extreme values are not possible. But there is nothing to stop the resulting \hat{N}_t predictions being more extreme than for the observed values of N_t , which includes allowing negative values. Negative values are predicted for six \hat{N}_t in our example time series (see File S1). We suggest the simple remedy of replacing the negative predictions with the smallest observed value from the original N_t time series. A second option is to replace N_t with $\log N_t$ (which can be negative), although results will differ because relative distances of nearest neighbours will change, altering the weights in (14); Rogers et al. (2022) implemented both N_t and $\log N_t$. A third option is to not first-difference the original data (discussed below).

Relatedly, we find $\rho = 0.83$, but calculating the correlation based on N_t and \hat{N}_t instead, by replacing Y with N in (15), gives 0.54; for $E = 2$ we get 0.70 and 0.28. Thus, we caution that high correlation based on Y_t does not necessarily imply high correlation based on N_t , which is what we are interested in (see below and File S1).

Condition (d) above is that it may be appropriate to exclude $\mathbf{x}_{t+1}, \mathbf{x}_{t+2}, \dots, \mathbf{x}_{t+E}$ from the library of candidate nearest neighbours of the focal point \mathbf{x}_t . This is based on the principle that when testing the predictive accuracy of a method it is problematic to use information about the value being predicted. The method should not have any knowledge of the known value of the quantity.

For our simulated data and $E = 2$, we find that predictions \hat{Y}_{t+1} are the same when using pbsEDM or rEDM, except for $t^* = 75$ (0.838 for pbsEDM versus 1.368 for rEDM) and $t^* = 94$ (0.412 versus 0.177). For $t^* = 75$, we find that rEDM uses $\mathbf{x}_{t+1} = \mathbf{x}_{76} = (Y_{76}, Y_{75})$ as one of the three nearest neighbours to \mathbf{x}_{75} , and hence uses it in the prediction \hat{Y}_{76} , despite it including Y_{76} (which is what we are trying to predict). We find this by changing the value of Y_{76} to a large value such that \mathbf{x}_{76} is no longer a close neighbour of \mathbf{x}_{75} , and the rEDM code then gives the exact same answer as for pbsEDM (also agreeing with some earlier code that we wrote independently of pbsEDM); see File S1.

Similarly, for $t^* = 94$ we find that rEDM uses $\mathbf{x}_{t+2} = \mathbf{x}_{96} = (Y_{96}, Y_{95})$ as a nearest neighbour of \mathbf{x}_{94} , but this neighbour includes the value

of Y_{95} that we are trying to predict (and we suggest it should be excluded). The default in rEDM is to not exclude any temporally adjacent neighbours, although the exclusionRadius argument allows the user to exclude nearest temporal neighbours within exclusionRadius time steps of t^* (i.e. this would exclude the exclusionRadius number of \mathbf{x}_t both above and below \mathbf{x}_t in (6), which can help deal with autocorrelation). For short time series as we have in our fisheries applications, we would like to retain as many potential neighbours as possible, and so in pbsEDM our default is as described above in (6) and (7), and we also provide options to match the settings from rEDM. Differences between such options will become more important for higher embedding dimensions than 2, since the excluded points $\mathbf{x}_{t+1}, \mathbf{x}_{t+2}, \dots, \mathbf{x}_{t+E}$, become more numerous as E increases.

Note that forecasting \hat{N}_{T+1} involves setting $t^* = T - 1$, for which the excluded points just referred to would be the undefined $\mathbf{x}_T, \mathbf{x}_{T+1}, \dots, \mathbf{x}_{T-1+E}$. So although the different options will not directly affect the nearest neighbours of \mathbf{x}_{T-1} and the \hat{N}_{T+1} calculation, they do affect the calculation of ρ and hence the choice of E used for forecasting, which can indeed influence \hat{N}_{T+1} .

For our simulated data the largest Y_t is Y_3 for $t = 3$ (Figure 1c). Predicting \hat{Y}_3 requires $t^* = 2$ which is valid for $E = 2$ but no higher E (Figure 4). Yet Y_3 is the poorest estimated value of all (being the right-most point of Figure 1f). So the most poorly estimated point is included in the ρ calculations only for $E = 2$, which seems an unfair constraint when comparing ρ for different E (Figure 3) to find the optimal E to use for forecasting. Future investigations could examine whether restricting calculations to the same set of t^* , based on Figure 4, should be done when determining the optimal E .

Whether to apply the first-differencing or not will be time-series dependent. Chang et al. (2017) state that linear trends in the original data should be removed, either by simple regression or taking the first-difference, to make the time series stationary. First-differencing was not strictly necessary for our example time series (there was no clear linear trend in the N_t), yet the first-differenced lagged values in Figure 1d do demonstrate geometric structure that is not seen in the non-first-differenced values in Figure 1b. Our explanations are the same without first-differencing, with Y_t simply taking the value N_t instead of $N_{t+1} - N_t$. Real applications can test sensitivity to first-differencing.

In Appendix S1 we extend the above mathematical description to the multivariate situation of analysing multiple variables, such as populations of several species or a population and an index of local temperature. The library of candidate nearest neighbours to the focal point can again be calculated. The size of the library does not depend on the chosen embedding dimension, just on the maximum lag, m , used for any of the variables. The size is once more represented by Figure 4, but with the E -axis replaced by $m + 1$. So the library size does not change if further variables are added unless they are lagged more than the existing variables such that m increases. We describe the S-map algorithm in Appendix S1 and apply it to our simulated data set.

5 | DISCUSSION

We have derived a thorough description of the core methods of EDM, yielding previously undocumented aspects that improve understanding. Having gained a deeper understanding of EDM, our work suggests potential enhancements. For example, the closest $E + 1$ neighbours are typically selected for the simplex algorithm (to form a simplex in the E -dimensional space), but simulations could investigate how altering the numbers of neighbours (an easily changed parameter in rEDM) might improve accuracy. This could lead to developing a bootstrapping approach to produce confidence intervals for simplex predictions. Simulation testing could determine the observed coverage of such intervals (and also for bootstrap intervals from the S-map algorithm, as used by Karakoç et al., 2020).

Readers searching the literature should be aware of other terms that describe EDM-type approaches, including nonlinear forecasting, state-space reconstruction, Takens' theorem, time-delay embedding and Jacobian Lyapunov exponents. To delve into the more technical background behind EDM, we recommend the books by Ott et al. (1994), particularly Chapter 5 on 'The Theory of Embedding' and the included reprints of Sauer (1993) and Sugihara and May (1990), and Huffaker et al. (2017), particularly Chapter 3 on 'Phase Space Reconstruction'.

The use of EDM can allow for time-varying productivity (or other ecosystem changes) to be implicitly accounted for in applications such as fisheries management. For example, Ye et al. (2015) found that including time series of sea surface temperature when forecasting salmon populations using EDM performed better than not including temperature, and that EDM outperformed parametric models. Fruitful research could further compare EDM with parametric time-varying models (for which it is necessary but hard to prescribe an explicit mathematical relationship between productivity and time). How this would directly inform decision-making requires further investigation, since, in general, accounting for nonstationarity in the ecosystem requires careful consideration of how to determine the benchmarks or reference points that are used to determine the status of stocks (Holt & Michielsens, 2020). So although we have described EDM as an alternative to parametric mechanistic modelling, both approaches can be used together in various complementary ways (Munch et al., 2020), and this may indeed be how EDM fulfils its potential in practical management applications.

AUTHOR CONTRIBUTIONS

Andrew M. Edwards: Conceptualization (equal); data curation (equal); formal analysis (lead); funding acquisition (equal); investigation (equal); methodology (equal); project administration (equal); software (equal); supervision (equal); validation (lead); visualization (lead); writing – original draft (lead); writing – review and editing (lead). **Luke Rogers:** Data curation (equal); formal analysis (supporting); investigation (equal); methodology (equal); software (equal); writing – review and editing (supporting). **Carrie Holt:** Conceptualization (equal); data curation (equal); formal analysis (supporting); funding acquisition (equal); investigation (equal); methodology (equal);

project administration (equal); supervision (equal); writing – original draft (supporting); writing – review and editing (supporting).

ACKNOWLEDGEMENTS

We thank Joe Watson, John Holmes and Travis Tai for useful comments on earlier drafts, plus three anonymous reviewers and the Editors for their comments that have improved this work. We are also grateful to George Sugihara, Hao Ye, Erik Saberski and Joseph Park for informative discussions. LAR thanks Fisheries and Oceans Canada for postdoctoral funding through FSERP (Fisheries Science and Ecosystem Research Program).

DATA AVAILABILITY STATEMENT

No new data were collected. All results are reproducible from the File S1.

ORCID

Andrew M. Edwards  <https://orcid.org/0000-0003-2749-8198>

Luke A. Rogers  <https://orcid.org/0000-0003-3843-064X>

Carrie A. Holt  <https://orcid.org/0000-0002-4699-9978>

REFERENCES

- Caswell, H., Fujiwara, M., & Braul, S. (1999). Declining survival probability threatens the North Atlantic right whale. *Proceedings of the National Academy of Sciences of the United States of America*, 96, 3308–3313.
- Chang, C.-W., Ushio, M., & Hsieh, C.-H. (2017). Empirical dynamic modeling for beginners. *Ecological Research*, 32, 785–796.
- Cleary, J. S., Hawkshaw, S., Grinnell, M. H., & Grandin, C. (2019). Status of B.C. Pacific Herring (*Clupea pallasii*) in 2017 and forecasts for 2018. *Fisheries and Oceans Canada. Canadian Science Advisory Secretariat Research Document*, 2018/028, v + 285 p.
- Deyle, E., Schueller, A. M., Ye, H., Pao, G. M., & Sugihara, G. (2018). Ecosystem-based forecasts of recruitment in two menhaden species. *Fish and Fisheries*, 19, 769–781.
- Deyle, E. R., Bouffard, D., Frossard, V., Schwefel, R., Melack, J., & Sugihara, G. (2022). A hybrid empirical and parametric approach for managing ecosystem complexity: Water quality in Lake Geneva under nonstationary futures. *Proceedings of the National Academy of Sciences of the United States of America*, 119, e2102466119.
- Deyle, E. R., Fogarty, M., Hsieh, C.-H., Kaufman, L., MacCall, A. D., Munch, S. B., Perretti, C. T., Ye, H., & Sugihara, G. (2013). Predicting climate effects on Pacific sardine. *Proceedings of the National Academy of Sciences of the United States of America*, 110, 6430–6435.
- Dorner, B., Peterman, R. M., & Su, Z. (2009). Evaluation of performance of alternative management models of Pacific salmon (*Oncorhynchus* spp.) in the presence of climatic change and outcome uncertainty using Monte Carlo simulations. *Canadian Journal of Fisheries and Aquatic Sciences*, 66, 2199–2221.
- Edwards, A. M. (2001). Adding detritus to a nutrient-phytoplankton-zooplankton model: a dynamical-systems approach. *Journal of Plankton Research*, 23, 389–413.
- Edwards, A. M., & Auger-Méthé, M. (2019). Some guidance on using mathematical notation in ecology. *Methods in Ecology and Evolution*, 10, 92–99.
- Fasham, M. J. R., Ducklow, H. W., & McKelvie, S. M. (1990). A nitrogen-based model of plankton dynamics in the oceanic mixed layer. *Journal of Marine Research*, 48, 591–639.
- Grziwotz, F., Strauß, J. F., Hsieh, C.-H., & Telschow, A. (2018). Empirical dynamic modelling identifies different responses of *Aedes*

- Polynesiensis* subpopulations to natural environmental variables. *Scientific Reports*, 8, 16768.
- Harford, W. J., Karnauskas, M., Walter, J. F., & Liu, H. (2017). Non-parametric modeling reveals environmental effects on bluefin tuna recruitment in Atlantic, Pacific, and Southern Oceans. *Fisheries Oceanography*, 26, 396–412.
- Holt, C. A., & Michielsens, C. G. J. (2020). Impact of time-varying productivity on estimated stock-recruitment parameters and biological reference points. *Canadian Journal of Fisheries and Aquatic Sciences*, 77, 836–847.
- Howell, D., Schueller, A. M., Bentley, J. W., Buchheister, A., Chagaris, D., Cieri, M., Drew, K., Lundy, M. G., Pedreschi, D., Reid, D. G., & Townsend, H. (2021). Combining ecosystem and single-species modeling to provide ecosystem-based fisheries management advice within current management systems. *Frontiers in Marine Science*, 7, 607831.
- Hsieh, C.-H., Glaser, S. M., Lucas, A. J., & Sugihara, G. (2005). Distinguishing random environmental fluctuations from ecological catastrophes for the North Pacific Ocean. *Nature*, 435, 336–340.
- Huffaker, R., Bittelli, M., & Rosa, R. (2017). *Nonlinear time series analysis with R*. Oxford University Press.
- Kantz, H., & Schreiber, T. (2004). *Nonlinear time series analysis*. Cambridge University Press.
- Karakoç, C., Clark, A. T., & Chatzinotas, A. (2020). Diversity and coexistence are influenced by time-dependent species interactions in a predator-prey system. *Ecology Letters*, 23, 983–993.
- May, R. M. (1976). Simple mathematical models with very complicated dynamics. *Nature*, 261, 459–467.
- Munch, S. B., Brias, A., Sugihara, G., & Rogers, T. L. (2020). Frequently asked questions about nonlinear dynamics and empirical dynamic modelling. *ICES Journal of Marine Science*, 77, 1463–1479.
- Munch, S. B., Rogers, T. L., & Sugihara, G. (2022). Recent developments in empirical dynamic modelling. *Methods in Ecology and Evolution*, 14, 732–745.
- Murray, J. D. (1989). *Mathematical Biology, volume 19 of Biomathematics*. Springer-Verlag.
- Ott, E., Sauer, T., & Yorke, J. A. (1994). *Coping with chaos: Analysis of chaotic data and the exploitation of chaotic systems*. Wiley series in nonlinear science. John Wiley and Sons, Inc.
- Packard, N. H., Crutchfield, J. P., Farmer, J. D., & Shaw, R. S. (1980). Geometry from a time series. *Physical Review Letters*, 45, 712–716.
- Park, J., & Smith, C. (2023). *pyEDM: Python wrapper for cppEDM using pybind11*. Python package version 1.14.0.2. <https://pepy.tech/project/pyEDM>
- Park, J., Smith, C., Sugihara, G., & Deyle, E. (2023). *rEDM: Empirical Dynamic Modeling (EDM)*. R package version 1.15.1. <https://CRAN.R-project.org/package=rEDM>
- Rogers, L. A., & Edwards, A. M. (2023). *pbsEDM: An R package to implement some of the methods of empirical dynamic modelling*. R package version 1.0.0. <https://github.com/pbs-assess/pbsEDM>
- Rogers, T. L., Johnson, B. J., & Munch, S. B. (2022). Chaos is not rare in natural ecosystems. *Nature Ecology & Evolution*, 6, 1105–1111.
- Sauer, T. (1993). Time series prediction using delay coordinate embedding. In A. S. Weigend & N. A. Gershenfeld (Eds.), *Time series prediction: Forecasting the future and understanding the past*, Santa Fe Institute studies in the science of complexity XV (pp. 175–194). Addison-Wesley.
- Schaffer, W. M., & Kot, M. (1986). Chaos in ecological systems: the coals that Newcastle forgot. *Trends in Ecology & Evolution*, 1, 58–63.
- Sguotti, C., Otto, S. A., Cormon, X., Werner, K. M., Deyle, E., Sugihara, G., & Möllmann, C. (2020). Non-linearity in stock-recruitment relationships of Atlantic cod: insights from a multi-model approach. *ICES Journal of Marine Science*, 77, 1492–1502.
- Stark, J., Broomhead, D. S., Davies, M. E., & Huke, J. (2003). Delay embeddings for forced systems. II. Stochastic forcing. *Journal of Nonlinear Science*, 13, 519–577.
- Sugihara, G. (1994). Nonlinear forecasting for the classification of natural time series. *Philosophical Transactions of the Royal Society of London, Series A*, 348, 477–495.
- Sugihara, G., May, R., Ye, H., Hsieh, C.-H., Deyle, E., Fogarty, M., & Munch, S. (2012). Detecting causality in complex ecosystems. *Science*, 338, 496–500.
- Sugihara, G., & May, R. M. (1990). Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. *Nature*, 344, 734–741.
- Takens, F. (1981). Detecting strange attractors in turbulence. In D. Rand & L. S. Young (Eds.), *Dynamical systems and turbulence, Warwick 1980, volume 898 of Lecture Notes in Mathematics* (pp. 366–381). Springer.
- Turchin, P. (2001). Does population ecology have general laws? *Oikos*, 94, 17–26.
- Ushio, M. (2022). Interaction capacity as a potential driver of community diversity. *Proceedings of the Royal Society B*, 289, 20212690.
- Van Beveren, E., Benoît, H. P., & Duplisea, D. E. (2021). Forecasting fish recruitment in age-structured population models. *Fish and Fisheries*, 22, 941–954.
- Wood, S. N., & Thomas, M. B. (1999). Super-sensitivity to structure in biological models. *Proceedings of the Royal Society of London. Series B*, 266, 565–570.
- Ye, H., Beamish, R. J., Glaser, S. M., Grant, S. C. H., Hsieh, C.-H., Richards, L. J., Schnute, J. T., & Sugihara, G. (2015). Equation-free mechanistic ecosystem forecasting using empirical dynamic modeling. *Proceedings of the National Academy of Sciences of the United States of America*, 112, E1569–E1576.
- Yool, A. (1998). *The dynamics of open-ocean plankton ecosystem models*. Ph.D. thesis, University of Warwick, U.K.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: Edwards, A. M., Rogers, L. A., & Holt, C. A. (2024). Explaining empirical dynamic modelling using verbal, graphical and mathematical approaches. *Ecology and Evolution*, 14, e10903. <https://doi.org/10.1002/ece3.10903>