IOP Publishing Mach. Learn.: Sci. Technol. 6 (2025) 025009 https://doi.org/10.1088/2632-2153/adc873

MACHINE LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
7 December 2024

REVISED
23 March 2025

ACCEPTED FOR PUBLICATION
1 April 2025

PUBLISHED
10 April 2025

PAPER

Robust prediction of chaotic systems with random errors using dynamical system deep learning

Zixiang Wu[1,2], Jianping Li[1,2,*], Hao Li[1,2], Mingyu Wang[1], Ning Wang[1] and Guangcan Liu[1]

[1] Frontiers Science Center for Deep Ocean Multi-spheres and Earth System (DOMES)/Key Laboratory of Physical Oceanography/Academy of Future Ocean/College of Oceanic and Atmospheric Sciences/Center for Ocean Carbon Neutrality, Ocean University of China, Qingdao 266100, People's Republic of China

[2] Laboratory for Ocean Dynamics and Climate, Qingdao Marine Science and Technology Center, Qingdao 266237, People's Republic of China

[*] Author to whom any correspondence should be addressed.

E-mail: ljp@ouc.edu.cn

Keywords: dynamical system deep learning, random errors, predictive robustness

Abstract

To predict nonlinear dynamical systems, a novel method called the dynamical system deep learning (DSDL), which is based on the state space reconstruction (SSR) theory and utilizes time series data for model training, was recently proposed. In the real world, observational data of chaotic systems are subject to random errors. Given the high nonlinearity and sensitivity of chaotic systems, the impact of random errors poses a significant challenge to the prediction. Mitigating the impact of random errors in the prediction of chaotic systems is a significant practical challenge. Traditional data-driven methods exhibit insufficient robustness against superimposed random errors, due to little consideration for temporal dynamic evolutionary of chaotic systems. Therefore, reducing the impact of random errors in the prediction of chaotic systems remains a difficult issue. In previous work, the DSDL demonstrated superiority in the noise-free scenario. This study primarily introduces the delay embedding theorem under noisy conditions and investigates the predictive capability of the DSDL in the presence of random errors in the training data. The performance of the DSDL is tested on three example systems, namely the Lorenz system, hyperchaotic Lorenz system and conceptual ocean–atmosphere coupled Lorenz system. The results show that the DSDL exhibits high accuracy and stability compared to various traditional machine learning methods and previous dynamic methods. Notably, as the magnitude of errors decreases, the advantage of the DSDL over traditional machine learning methods becomes more pronounced, highlighting the DSDL's capacity to effectively extract the temporal evolution characteristics of chaotic systems from time series and to identify the true system state within observational error bands, significantly mitigating the impact of random errors. Moreover, unlike other contemporary deep learning methods, the DSDL requires faster hyperparameter tuning by using fewer parameters for improving accuracy, and based on the advantage of the SSR theoretical framework, the DSDL does not require prior knowledge of the original governing equations. Our work extends the theoretical applicability of the DSDL under random error conditions and points to the new and superior data-driven method DSDL based on the dynamic framework, holding significant potential for mitigating the impact of random errors and achieving robust predictions of real-world systems.

1. Introduction

Applications and research on chaotic systems are widespread across various disciplines, such as ecology [1], finance [2], climatology [3], neuroscience [4], and cryptography [5–7]. Due to the complex dynamics, predicting chaotic systems is considered a challenging task [8]. Leveraging the dynamical information inherent in nonlinearity is crucial for effective predictions [9]. Nonlinearity can be explicitly represented in

equations or may remain hidden within time series data [10, 11]. Exploiting the hidden dynamic interaction information from available time series for predictive purposes is an important issue [12].

The state space reconstruction (SSR) theory marks a significant advancement in understanding chaotic dynamics [13, 14]. The SSR originates from the Takens' delay embedding theorem, which enables the reconstruction of a phase space that is topologically equivalent to the original system through delayed-coordinates of univariate time series [15–17]. Numerous studies have expanded the SSR to more general cases, particularly in multivariate reconstructions [18–20]. Ma *et al* introduced the principle of inverse delay embedding, employing the improved SSR to predict high-dimensional chaotic dynamical systems [21, 22]. The SSR is increasingly applicable to real-world chaotic prediction scenarios [23, 24].

In the era of big data, deep learning has flourished in different fields [25], such as stock prediction [2, 26], weather forecast [27, 28], video encryption [29]. Many methods demonstrate remarkable performance in chaotic prediction through extensive training on historical data, such as the long short-term memory network (LSTM) [30, 31], convolution neural network [31–33], reservoir computing [34, 35]. However, deep learning fundamentally operates as a statistical method that often lacks integration with dynamical theory, leading to an insufficient understanding of the intrinsic properties of chaotic systems [36–38]. Recently, there has been a growing focus on physics-informed data-driven methods that emphasize physical relationships in predictions [39, 40]. By combining the SSR with data-driven methods, previous work proposed the dynamical system deep learning (DSDL), which effectively leverages the dynamic information underlying the sufficient data and enhances the application of SSR in chaotic prediction [41, 42]. The DSDL outperforms traditional deep learning methods in prediction accuracy and exhibits a degree of interpretability.

However, in the real world, most available time series are derived from observational data, which often contains random errors that affect the accuracy of various methods in chaotic system prediction [22, 35]. Chaotic systems are characterized by high nonlinearity and sensitivity, where even minute changes in initial conditions or parameters can lead to significant deviations in system behavior [43]. Consequently, the impact of random errors poses a critical challenge in the prediction of chaotic systems. Random errors, typically arising from observational inaccuracies or external disturbances, can be rapidly amplified during the prediction process. This cumulative effect of prediction errors results in substantial deviations between long-term predictions and the true system states. For instance, in meteorology, minor observational errors can lead to the complete failure of weather and climate predictions [3, 44]. Similarly, in chaotic systems modeling financial markets, the interference of random noise may render the prediction model incapable of accurately capturing market trends [2, 26]. Traditional data-driven methods, while partially accounting for nonlinearity, often fail to fully consider the temporal dependency and evolutionary properties of chaotic systems, limiting their ability to identify true system states [45, 46], e.g. the LSTM [47], echo state networks (ESNs) [48]. As a result, these methods exhibit insufficient robustness against superimposed random errors. Reducing the influence of random errors remains a practical and difficult issue.

The DSDL method, grounded in the SSR theoretical framework, thoroughly considers the temporal evolution characteristics of chaotic systems. In previous work, the DSDL model was applied to noise-free scenarios and demonstrates superiority in predicting chaotic systems [41, 42]. Though random errors may have influence on the SSR [49, 50], the stable Takens' delay embedding theorem was proposed, indicating that the SSR remains feasible under conditions of random errors [51, 52]. There has been relatively little research on applying the SSR theory under noisy conditions to the prediction of noisy chaotic systems. Introducing the noisy SSR theory provides a theoretical foundation for the DSDL method in predicting chaotic systems with random errors. Therefore, the DSDL, as a data-driven method with dynamic framework, holds significant potential for enhancing predictive capabilities when analyzing noisy chaotic system data.

In this study, we seek to examine whether the DSDL prediction model remains feasible and effective when data contains random errors and whether the DSDL continues to maintain superior predictive capability. The structure of this paper is as follows: section 2 introduces the DSDL framework and the experimental conditions. Section 3 demonstrates the effectiveness of the DSDL in the presence of random errors, along with comparisons between the DSDL and other contemporary methods. Section 4 concludes with the key findings of this study and outlines future research directions.

2. Methods

2.1. Benchmark systems

In order to obtain datasets for training and testing, we employ the fourth-order Runge–Kutta scheme (RK4) with a time step $\Delta t = 0.01$ in the dimensionless time units (DTUs) to integrate forward for 10^7 steps, starting from specified initial conditions. The first 5000 time points, i.e. the first 50 DTU, in numerical solution are discarded for ensuring the chaos of the time series. The predictive capacities of the DSDL model and other methods are tested on the following three dynamical chaotic systems.

2.1.1. Lorenz system

The Lorenz system is the well-known 3-dimension strange attractor representing chaotic characteristic of the fluid layer [43]. The Lorenz system is described by the three coordinates:

$$\begin{cases} \frac{\mathrm{d}x}{\mathrm{d}t} = \sigma(y-x) \\ \frac{\mathrm{d}y}{\mathrm{d}t} = rx - y - xz \\ \frac{\mathrm{d}z}{\mathrm{d}t} = xy - bz \end{cases}, \tag{1}$$

where the parameters are $(\sigma,\ r,\ b) = (10,\ 24,\ 8/3)$. The numerical solution is integrated starting from the initial condition $(x,\ y,\ z) = (0,\ 1,\ 0)$.

2.1.2. Hyperchaotic Lorenz system

The hyperchaotic Lorenz system was proposed through introducing an additional state variable x_4 and coupling with the Lorenz system [53], which is expressed as:

$$\begin{cases} \frac{\mathrm{d}x_1}{\mathrm{d}t} = a(x_2 - x_1) \\ \frac{\mathrm{d}x_2}{\mathrm{d}t} = bx_1 + cx_2 - x_1x_3 + x_4 \\ \frac{\mathrm{d}x_3}{\mathrm{d}t} = -\mathrm{d}x_3 + x_1x_2 \\ \frac{\mathrm{d}x_4}{\mathrm{d}t} = -kx_1 \end{cases}, \tag{2}$$

where the parameters are set to be $(a,\ b, c,\ d,\ k) = (35,\ 7,\ 12,\ 3,\ 5)$. The initial condition is $(x_1, x_2, x_3, x_4) = (0,\ 1,\ 0,\ 0)$ for integrating numerical solution.

2.1.3. Conceptual ocean–atmosphere coupled Lorenz system

To reflect the coupling processes between the atmosphere and the ocean, a slowly-varying upper ocean variable w and a deep ocean variable η were added in the Lorenz system [54]. Describing the conceptual ocean–atmosphere coupled model as a 5-variable coupled model, the equations are expressed as:

$$\begin{cases} \frac{\mathrm{d}x}{\mathrm{d}t} = \sigma(y-x) \\ \frac{\mathrm{d}y}{\mathrm{d}t} = (1+c_1w)\kappa x - y - xz \\ \frac{\mathrm{d}z}{\mathrm{d}t} = xy - bz \\ O_m\frac{\mathrm{d}w}{\mathrm{d}t} = c_2y + c_3\eta + c_4w\eta - O_\mathrm{d}w \\ \qquad\qquad +S_m + S_s\cos\left(2\pi t/S_{\mathrm{pd}}\right) \\ \Gamma\frac{\mathrm{d}\eta}{\mathrm{d}t} = c_5w + c_6w\eta - O_\mathrm{d}\eta \end{cases}, \tag{3}$$

where the parameters are set as:

$$\begin{pmatrix} \sigma, \kappa, b, c_1, c_2, O_m, O_d, S_m, \\ S_s, S_{pd}, \Gamma, c_3, c_4, c_5, c_6 \end{pmatrix} = \begin{pmatrix} 9.95,\ 28,\ 8/3,\ 10^{-1},\ 1,\ 10,\ 1,\ 10, \\ 1,\ 10,\ 100,\ 10^{-2},\ 10^{-2},\ 1,\ 10^{-3} \end{pmatrix}, \tag{4}$$

and initial condition is $(x,\ y,\ z,\ w,\ \eta) = (0,\ 1,\ 0,\ 0,\ 0)$.

2.2. The DSDL framework

For an n dimensional chaotic system, there are n primitive variables with corresponding time series $x_i(t)$, $i = 1, 2, \cdots, n$. To predict the evolution of the chaotic system, each primitive variable may serve as a prediction target. Here, for $i = k$, $x_k(t)$ is the target variable, where $k \in [1, n]$. The main structure of the DSDL framework is listed as follows:

(1) According to the SSR theory, both univariate and multivariate forms can reconstruct the primitive chaotic system using delayed embedding and generalized embedding, respectively. The attractors are derived from the reconstruction of univariate and multivariate forms, referred to as 'delayed attractors' $\mathcal{D}$ [15–17] and 'non-delayed attractors' $\mathcal{N}$ [18, 19], which are related through a mapping relationship $\boldsymbol{\Phi}$. This mapping connects the delayed states of a single variable with the non-delayed states of multiple variables, implying that the univariate reconstruction contains states corresponding to future moments relative to the

multivariate reconstruction. The mapping $\boldsymbol{\Phi}$ is expressed as:

$$\boldsymbol{\Phi}:\begin{pmatrix} x_i(t) & x_i(t+\tau) & \cdots & x_i(t+(m-1)\tau) \\ x_j(t) & x_j(t+\tau) & \cdots & x_j(t+(m-1)\tau) \\ x_s(t) & x_s(t+\tau) & \cdots & x_s(t+(m-1)\tau) \\ \vdots & \vdots & \cdots & \vdots \\ x_p(t) & x_p(t+\tau) & \cdots & x_p(t+(m-1)\tau) \end{pmatrix} \rightarrow \begin{pmatrix} x_k(t) & x_k(t+\tau) & \cdots & x_k(t+(m-1)\tau) \\ x_k(t+\tau) & x_k(t+2\tau) & \cdots & x_k(t+m\tau) \\ x_k(t+2\tau) & x_k(t+3\tau) & \cdots & x_k(t+(m+1)\tau) \\ \vdots & \vdots & \cdots & \vdots \\ x_k(t+(N-1)\tau) & x_k(t+N\tau) & \cdots & x_k(t+(N+m-2)\tau) \end{pmatrix}, \tag{5}$$

where $(x_i, x_j, x_s, \cdots, x_p)$ are multiple variables, and $i, j, s, \cdots, p \in [1, n]$. Notably, N in the mapping (5) also represents the number of multiple variables introduced in the non-delayed attractors. Thus, the mapping expression between the two reconstruction attractors is simplified as $\boldsymbol{\Phi}: \mathcal{N} \rightarrow \mathcal{D}$.

Through the fitting process given by:

$$\widetilde{\varphi_k}(\mathcal{N}) = \begin{pmatrix} x_k(t+\tau) & x_k(t+2\tau) & \cdots & x_k(t+m\tau) \end{pmatrix}, \tag{6}$$

where m is the data length for fitting, we can establish the mapping relationship $\widetilde{\varphi_k}$ between multiple variables and the target variable x_k. After fitting $\widetilde{\varphi_k}$, the prediction model can be expressed as:

$$\widetilde{x_k}(t+(m+1)\tau) = \widetilde{\varphi_k}\begin{pmatrix} x_i(t+m\tau) & x_j(t+m\tau) & \cdots & x_p(t+m\tau) \end{pmatrix}, \tag{7}$$

which utilizes the state of multiple variables at the previous time to predict the state of the target variable at the next time.

In this study, time interval $\tau = 0.01$ DTU, which is equal to the time step Δt used in the numerical solution for the benchmark chaotic systems. The training data length is set as $m = 10\,000$ time points, i.e. 100 DTU, for ensuring sufficient model training.

(2) The predictor set $\boldsymbol{X_N}$ is defined as the set of variables introduced in the non-delayed attractor $\mathcal{N}$. In order to predict a dynamical chaotic system, it is essential to consider the nonlinear interactions among variables. Through multiplying linear variables in the primitive system by each other, the nonlinear variables are introduced into $\boldsymbol{X_N}$, which encapsulate information about these interactions. Based on the orders of the variables in $\boldsymbol{X_N}$, we can construct a multi-layer nonlinear network [41].

However, not all linear or nonlinear variables contribute positively to the predictions. Therefore, it is crucial to select the variables that significantly influence the temporal evolution of the dynamical chaotic system and enhance prediction accuracy. These variables are referred to as key variables. From $\boldsymbol{X_N}$, the key variable set $\boldsymbol{X_{K,x_k}}$ can be selected for the target variable $x_k(t)$ using the cross-validation-based stepwise regression (CVSR) [55]. The detailed steps of the CVSR are given as the flowchart (figure 10). The main process of CVSR involves sequentially introducing the variable from the candidate set that most reduces the model's fitting error, continuing until the introduction of any remaining candidate variable no longer significantly reduces the fitting error. If the introduction of a certain variable does not significantly reduce the fitting error during the fitting phase, that variable is considered ineffective in improving the model's fitting performance. There are inevitably interference variables in the candidate set that are unrelated to the system's evolution. These variables do not contribute effectively and will not be selected into the key variable set $\boldsymbol{X_{K,x_k}}$ theoretically. We construct the candidate variable set $\boldsymbol{X_N}$ into the multi-layer nonlinear network and select key variables layer by layer through the CVSR (shown in figure 9 as Selection mode). Consequently, we introduce the key variable set $\boldsymbol{X_{K,x_k}}$ to replace the original predictor set $\boldsymbol{X_N}$, thereby enhancing the foundational prediction model and establishing a more interpretable DSDL model.

The main architecture of DSDL is also given in figure 8. The mapping $\boldsymbol{\Phi}$ provides a solid predictive relationship between multivariate inputs and univariate outputs, which can be written as formula:

$$\begin{pmatrix} a_{k,1} & \cdots & a_{k,N} \end{pmatrix}\begin{pmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{Nm} \end{pmatrix} = \begin{pmatrix} x_{k,2} & \cdots & x_{k,m+1} \end{pmatrix}, \tag{8}$$

for predicting any one system variable x_k. Leveraging the predictive relationship, the fundamental predictive framework of the DSDL is established. The coefficient vector, which represents the theoretical mapping $\boldsymbol{\Phi}$, exactly is being fit for predicting each system variable $\left(\widetilde{x_{k,m+2}}, \widetilde{x_{k,m+3}}, \cdots\right)$. Multiple variables $x_1, \cdots, x_N$ should be selected as the key variables.

We introduce the detailed steps of the DSDL, as follows:

Step 1. Data input. For a n-dimensional dynamical system observational data, given training length m, the inputting data is donated as $X_{n\times m}$ matrix.

Step 2. Construct candidate variable set $\boldsymbol{X_N}$. Introduce nonlinear variables by the form of monomials, and then provide $X_{N\times m}$ matrix composed of N candidate variables. According to the order of the variables, $\boldsymbol{X_N}$ can be expressed as

$$\boldsymbol{X_N} = \left[\boldsymbol{X}, \boldsymbol{X^{(2)}}, \boldsymbol{X^{(3)}}, \ldots, \boldsymbol{X^{(l)}}\right], \tag{9}$$

where the highest order of the candidate variables is l. We define that $\boldsymbol{X_N}$ is composed of l-layers nonlinear network.

Step 3. For each primitive variable x_i, $i = 1, 2, \cdots, n$, select key variables $\boldsymbol{X}_{\boldsymbol{K},x_i}$ layer by layer using the CVSR. According to the mapping $\boldsymbol{\Phi}$, fit the expression $A_i^{l_1} X^{l_1}_{N_{l_1}\times(1:m-1)} = X_{i(2:m)}$ by multiple linear regression from the 1st-layer variables, which can be written as

$$\begin{pmatrix} a^{l_1}_{i,1} & \cdots & a^{l_1}_{i,N_{l_1}} \end{pmatrix} \begin{pmatrix} x^{l_1}_{11} & \cdots & x^{l_1}_{1(m-1)} \\ \vdots & \ddots & \vdots \\ x^{l_1}_{N_{l_1}1} & \cdots & x^{l_1}_{N_{l_1}(m-1)} \end{pmatrix} = \begin{pmatrix} x_{i,2} & \cdots & x_{i,m} \end{pmatrix}, \tag{10}$$

where l_1 represents the 1st-layer variables. Through the CVSR, the key variables set from the 1st-layer variables can be selected as $\boldsymbol{X}^{l_1}_{\boldsymbol{K},x_i}$.

Step 4. Key variables selected in the previous layer are inputted to the next layer. The 2nd-layer variables include $\boldsymbol{X}^{l_1}_{\boldsymbol{K},x_i}$. Similarly, according to $A_i^{l_2} X^{l_2}_{N_{l_2}\times(1:m-1)} = X_{i(2:m)}$, the key variables set from the 2nd-layer variables can be selected as $\boldsymbol{X}^{l_2}_{\boldsymbol{K},x_i}$ using the CVSR. Repeat the selection till to the lth-layer variables and obtain $\boldsymbol{X}^{l_l}_{\boldsymbol{K},x_i}$ as the key variables $\boldsymbol{X}_{\boldsymbol{K},x_i}$ for the primitive variable x_i.

Step 5. Predict one step. For each primitive variable x_i, $i = 1, 2, \cdots, n$, assume the number of key variables in $\boldsymbol{X}_{\boldsymbol{K},x_i}$ is k_i. The key variables matrix is donated as X_{K_i}. Fit the coefficient vector $\widetilde{A_{K_i}}$ by the expression $\widetilde{A_{K_i}} X_{K_i,k_i\times(1:m-1)} = X_{i(2:m)}$. Then, predict the next step by $\tilde{X}_{i(m+1)} = \widetilde{A_{K_i}} X_{K_i,k_i m}$. And obtain the n-dimensional prediction vector $\tilde{X}_{(1:n)(m+1)}$.

Step 6. Predict the next step continuously in iteration. According to the key-variable nonlinear combination, obtain the new input $\tilde{X}_{K_i,(1:k_i)(m+1)}$ and predict the next step by $\tilde{X}_{i(m+2)} = \widetilde{A_{K_i}} \tilde{X}_{K_i,(1:k_i)(m+1)}$. Repeat the prediction way till to specified prediction steps.

The detailed steps of the DSDL above have been summarized in a flowchart (figure 9).

2.3. Experimental conditions for introducing random errors

For an L-dimension dynamical system, we introduce random errors. Let $\{\xi_n\}$ represent a sequence of independent identically distributed L-dimension random vectors. The random vector $\xi_i(t)$ denotes a time-varying L-dimensional random error vector corresponding to L variables of the dynamical system. Each ξ_i in $\{\xi_n\}$ follows an approximately normal distributions, denoted as $\xi_i(t) \sim N\left(0, \sigma'^2\right)$, where σ' is an L-dimensional vector representing the standard deviation of the random errors associated with the L variables.

Random errors reflect the difference between observational data and the true state of the dynamical system. The introduction of random errors in the system can be expressed as:

$$X_O(t) - X_R(t) = \xi_i(t), \tag{11}$$

where $X_R(t)$ is the true state and $X_O(t)$ represents the observational data. In practice, $X_O(t)$ is the available data used for model training, which can be expressed as:

$$X_O(t) = X_R(t) + \xi_i(t). \tag{12}$$

In the real world, observational errors are present, represented here by the random error $\xi_i(t)$. Observations cannot be entirely accurate, as both the true values of the data and the random errors remain unknown.

To investigate the impact of random errors on prediction, we conduct tests on deterministic chaotic dynamical systems and consider the numerical solution as the true state of the systems. To define the true state, we assume that the numerical solution is derived from an ideal model, meaning that parameter errors and initial errors are not accounted for and are always considered to be zero.

To clearly facilitate the description of different error introduction scenarios, we distinguish between the introduction of random errors in the training set and the prediction set, as follows:

$$X_O^T(t) - X_R^T(t) = \xi_i^T(t), \tag{13}$$

$$X_O^P(t) - X_R^P(t) = \xi_i^P(t), \tag{14}$$

where the superscripts T and P represent the data during the training period and the prediction period, respectively. Three kinds of error introduction scenarios (figure 11) are studied for a dynamical system as follows:

(1) The ideal scenario is the absence of observational errors, represented as $\xi_i^T \equiv 0$ and $\xi_i^P \equiv 0$. In this scenario, the observational data is equivalent to the true data, and theoretical solutions are ideally considered as the observations.

(2) By introducing initial errors in the prediction test sets, the conditions can be described as $\xi_i^P(0) \neq 0$, while for $t > 0, \xi_i^P(t) \equiv 0$ and $\xi_i^T(t) \equiv 0$. This scenario indicates that the data is accurate in the training sets, with observational errors introduced only at the start of the prediction. To quantify the magnitude of the initial error, we define ε_0 and let $X_O^P(0) = X_R^P(0)(1+\varepsilon_0)$. Therefore, the initial error can be given as:

$$\xi_i^P(0) = \varepsilon_0 X_R^P(0), \tag{15}$$

where ε_0 is a relative error magnitude for representing the magnitude of the initial error.

(3) Random errors are present in the entire observational dataset. The condition is defined as $t \geqslant 0$, $\xi_i(t) \equiv 0$. This scenario closely resembles reality, and we primarily conduct experiments under these conditions to investigate whether the DSDL model can make accurate predictions in the presence of random errors throughout the entire training set. To quantify the magnitude of these random errors, we define ε as the ratio between the standard deviation of the random errors (donated as σ') and the standard deviation of the original dynamical system (donated as σ'), as follows:

$$\sigma' = \varepsilon\sigma, \tag{16}$$

$$\xi_i(t) \sim N\left(0, (\varepsilon\sigma)^2\right), \tag{17}$$

where $\xi_i(t)$ are specifically assumed to follow a normal distribution, both σ' and σ are L-dimensional vector. The metric ε is defined as a relative error magnitude that quantifies the magnitude of the random errors. To clearly describe the magnitude of random errors relative to the true data, we normalize the units of relative error ε as percentage.

2.4. Assessment of predictive capability

To quantify the solving and predicting capabilities of various methods, we use the metric of effective prediction time (EPT). For the time series $X(t)$ of a dynamical system, any method predicts the time series denoted as $\tilde{X}(t)$. Thus, the prediction error of the method also varies with time, expressed as:

$$E(t) = \left|X(t) - \tilde{X}(t)\right|. \tag{18}$$

An EPT is defined as the continuous prediction time during which the prediction error does not exceed the tolerance error [41], given as:

$$\mathrm{EPT} = \mathrm{Time}\left[E(t) \leqslant \delta\sigma\right], \tag{19}$$

where σ is the standard deviation of the dynamical system and δ is set as a relative metric to represent the tolerance error magnitude. The tolerance error is utilized to identify the point at which the model's accurate predictions cease. In this study, the tolerance error is set as $\delta = 1.0$, as the predicted sequence diverges significantly from the original sequence, leading to a loss of accuracy in subsequent predictions [42]. The higher the EPT, the better the method performs for the prediction of the dynamical chaotic system. Notably, the EPT here is in the DTUs.

To ensure reliable results, we primarily utilize the mean EPT to compare the predictive capabilities of different methods. The mean EPT is calculated as the mean of EPTs obtained from 100 distinct training and prediction periods. We normalize the EPT based on the Lyapunov time of each dynamical system.

3. Results

3.1. Effective prediction model with random errors

To demonstrate the effectiveness of the DSDL method in the presence of random errors, we establish the prediction model across three different chaotic dynamical systems (see Methods), each with distinct dimensions and incorporating random errors. The three dynamical systems are derived from the classical Lorenz system and exhibit typical chaotic properties, making them representative examples for the theoretical numerical experiments in this study. First, we need to identify appropriate candidate variable sets for each of the three chaotic systems. In the DSDL method, constructing a multi-layer nonlinear network (see Methods) requires determining the optimal highest order of the introduced variables based on data sample sets [41]. This step is intended to determine the number of nonlinear layers needed for the candidate variable sets. With the introduction of random errors, such as a relative error magnitude $\varepsilon = 0.01\%$, the predictive capabilities of the DSDL model vary with the number of nonlinear layers across different chaotic dynamical systems (figures 1(a)–(c)). The different colored lines represent the mean EPT, obtained from 100 distinct training and prediction test sets for each independent variable in the original chaotic systems. For the three chaotic dynamical systems analyzed, the optimal number of nonlinear layers is found to be 4. This indicates that the highest order of nonlinear variables introduced in the candidate variable set is 4, which enables the DSDL model to achieve optimal predictive performance in the three chaotic dynamical system with the relative error magnitude $\varepsilon = 0.01\%$. Furthermore, introducing higher-order variables does not significantly enhance the predictive capabilities of the DSDL model, while it does increase the associated computational costs.

After determining the number of nonlinear layers, we can select the key variable sets using the CVSR and build the DSDL predictive model based on training data. Figure 2 illustrates the predictions of the DSDL models for the three chaotic dynamical systems subjected to random errors. Despite the presence of the random errors, the DSDL models demonstrate reliable predictive capabilities. The accurate and continuous predictions for each primitive variable highlight the effectiveness of the DSDL models in handling random errors. Furthermore, the prediction trajectory consistently remains within the state space of the attractor for the Lorenz system, irrespective of whether the prediction periods are effective or ineffective (figures 2(a) and (b)), indicating that the DSDL model preserves the chaotic properties throughout the prediction. The conclusion also applies to the hyperchaotic Lorenz system and conceptual ocean–atmosphere coupled Lorenz system (figures 2(c) and (d)). Through the establishment of each DSDL prediction model, the future states of all components in each chaotic system can be effectively predicted under random errors. Notably, in the conceptual ocean–atmosphere coupled Lorenz system, time-varying factors are considered, i.e. the governing parameter of variable w in the equation (3) is not constant but time-varying, exhibiting time-dependent dynamics. The DSDL method remains reliable for predicting such time-varying dynamical systems under random errors. Furthermore, for slowly varying process variables in the system, e.g. ocean variables w and η, the predictive performance of the DSDL model is superior and less sensitive to random errors.

To comprehensively demonstrate the capability of the DSDL prediction model in handling random errors, we calculate the mean EPTs based on 100 different training and test sets. The training length is set as 100 DTUs, i.e. 10 000 time points. The mean EPTs are mainly used to compare the predictive performance of the DSDL model with that of other traditional machine learning methods (e.g. support vector regression (SVR) [56], vector autoregressive model (VAR) [57], autoregressive model (AR) [58]) and other contemporary deep learning methods (e.g. ESNs [34], LSTM [30], next generation reservoir computing (NG-RC) [59]) across the three chaotic dynamical systems with random errors. Firstly, we consider a relative error magnitude as $\varepsilon = 0.01\%$ (figure 3(a)). According to the formulas (16) and (17), the case of relative error $\varepsilon = 0.01\%$ means adding random errors $\xi_i(t)$ on the original training sets, where the ratio between the standard deviation of the random errors and that of the original dynamical system is set as 0.01%. According to the standard deviation of each chaotic systems, the training sets containing random errors is given to test the prediction capability of different methods in the three chaotic systems. In comparing the different predictive methods, we use the predictions of the first variable from the equations (1)–(3) to represent the overall predictive capability of each method for the target chaotic system. The DSDL model clearly excels in predictive accuracy compared to other methods, both in the noise-free scenarios [41] and in cases where the chaotic systems incorporate random errors.

To further investigate the predictive stability of the DSDL across different training/test sets with random errors, we compare the DSDL and other methods using the coefficient of variation while assessing model performance through the mean EPT (figure 3(c)). The dots representing the prediction results of the DSDL model for the chaotic systems lie within the pink shadow, indicating that the DSDL EPTs obtained from different training/test sets show greater overall prediction accuracy and lower dispersion. A higher the mean EPT and a lower the coefficient of variation indicates better performance. The DSDL model demonstrates

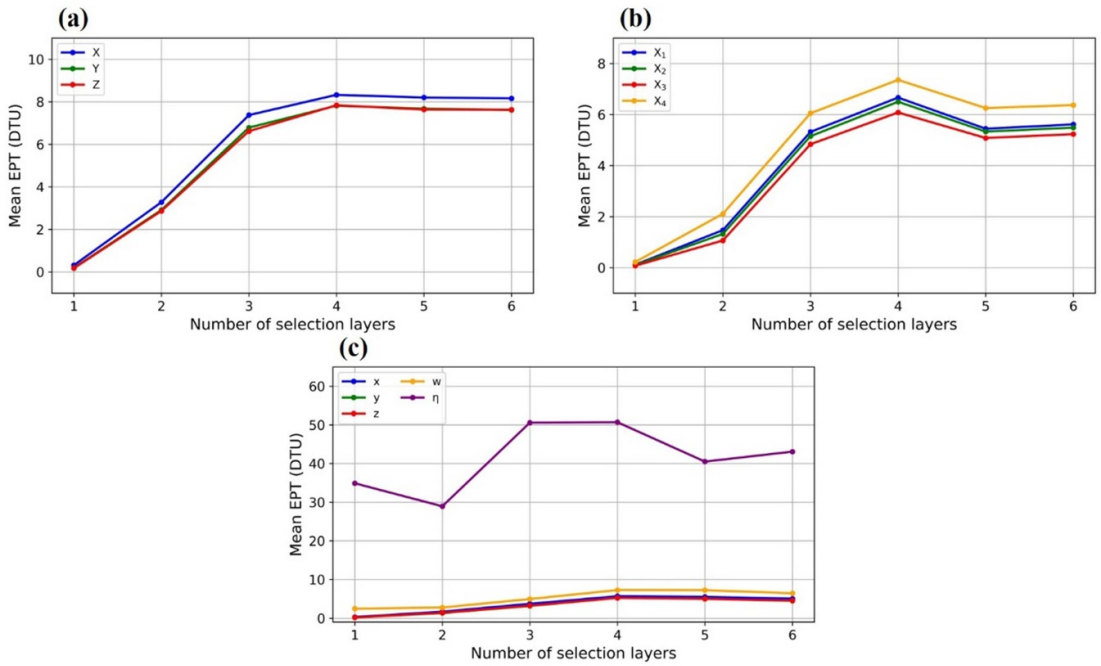

Figure 1. The effect of different numbers of nonlinear layers on the model predictive capability in three different chaotic dynamical systems with random errors. (a) The Lorenz system. The red, blue and green lines represent the variables x, y and z of the Lorenz system, respectively. The mean effective prediction time (EPT) is obtained by 100 different training and prediction test sets with random errors in which a relative error magnitude is $\varepsilon = 0.01\%$ to quantify the model predictive capability. The mean EPT is in dimensionless time units (DTUs). (b) Same as (a), but for the hyperchaotic Lorenz system. (c) Same as (a), but for the conceptual ocean–atmosphere coupled Lorenz system.

greater accuracy and stability in the presence of random errors compared to traditional machine learning methods. When the relative error magnitude of random errors increases to $\varepsilon = 1\%$, the results similarly illustrate the superior predictive capability of the DSDL method, as previously described (figures 3(b) and (d)).

For a prediction method, computational efficiency is also an important performance metric to consider, second only to prediction accuracy. We analyze the computational efficiency of the DSDL and other contemporary deep learning methods by measuring the CPU time and memory usage, which are used to indicate runtime and computational complexity respectively. We distinguish the computational efficiency of different methods during both the training and prediction periods, as tables 1 and 2. During the training period, the computational efficiency of the DSDL shows a moderate level (table 1). It decreases with the system's increasing dimensionality but remains within an acceptable range. Moreover, the computational cost required for hyperparameter optimization is not yet reflected in the quantitative metrics. Traditional deep learning methods typically require consideration of hyperparameter optimization. During the hyperparameter experiments, the computational complexity increases significantly. In contrast, the DSDL framework does not employ traditional neural network, and therefore, it does not involve hyperparameters typically associated with neural network optimization. Through faster hyperparameter tuning by using fewer parameters, the DSDL achieves its predictive capabilities by leveraging the SSR theory framework and key variable selection process, which ensures that the DSDL is capable of identifying and extracting the true evolutionary relationships intrinsic to the dynamical system within the error bands. The hyperparameters may require continuous optimization and adjustment to handle unknown random errors. Faster hyperparameter tuning by using fewer parameters suggests the DSDL training costs are not cumbersome and the DSDL demonstrates robustness to the impact of random errors.

As shown in table 2, once the DSDL model is built, the computational cost during the prediction period is significantly reduced. Compared to the traditional deep learning methods, such as the ESN and LSTM, the DSDL requires minimal computational cost during the prediction period. This indicates that while the DSDL needs a certain computational cost during the training period, the prediction accuracy of the established DSDL model is superior (shown in figure 3), and the computational efficiency during the prediction period demonstrates a certain advantage. The significant reduction in computational cost from training to prediction exhibited by the DSDL indicates the high efficiency and convenience of the DSDL predictive model once it has been adequately trained, highlighting the advantages of the DSDL predictive

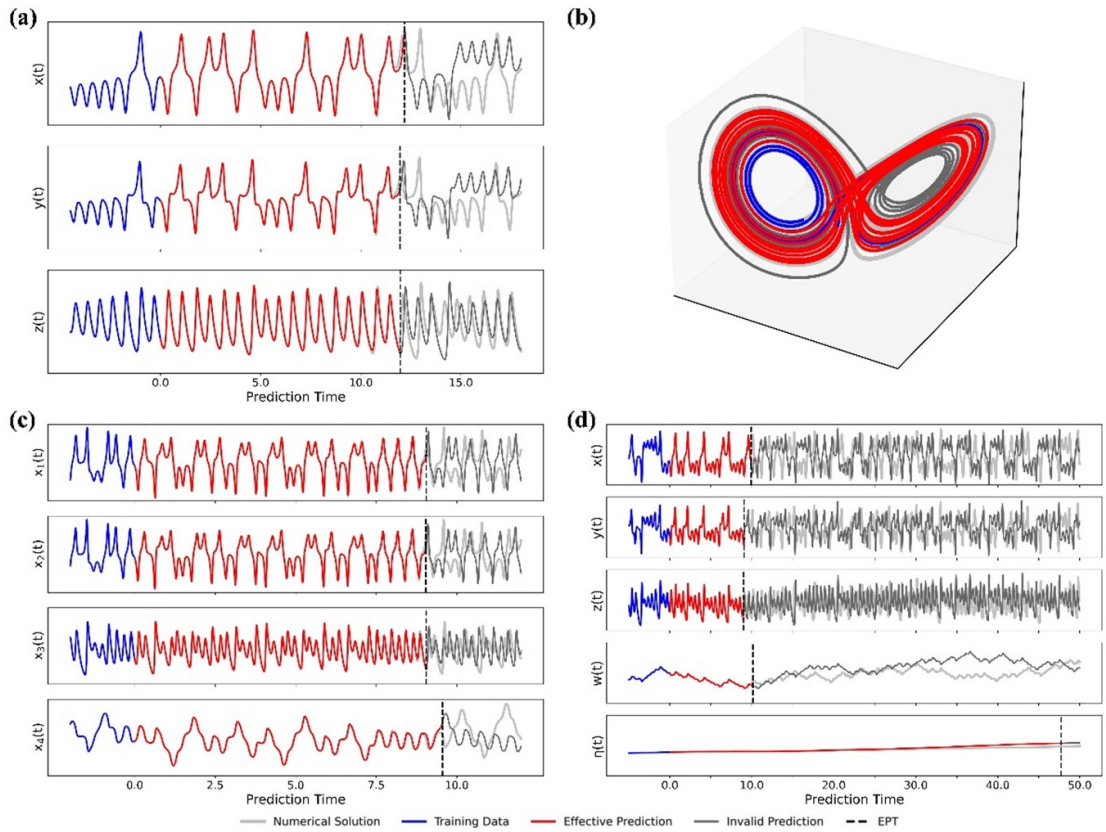

Figure 2. Predictions of the DSDL models for three chaotic dynamical systems with random errors. (a) The prediction for each variable of the Lorenz system. The light grey, blue, red and dark grey lines show the numerical solutions (true state), training set, effective predictions and invalid predictions in the corresponding prediction test set, respectively. The vertical black dashed line marks the EPT. The prediction time is in DTUs. Using a training set of 10^4 time points with a time step $\Delta t = 0.01$ DTU, i.e. 100 DTU, here only the last 500 time points are shown in this figure. The training set adds a sequence of random errors with a relative error magnitude $\varepsilon = 0.01\%$. (b) The prediction trajectory of the Lorenz attractor in the phase space. The prediction time is the same as the one in (a). (c) Same as (a), but for the hyperchaotic Lorenz system. (d) Same as (a), but for the conceptual ocean–atmosphere coupled Lorenz system.

framework. Through the comparative analysis of prediction accuracy and computational costs, the superiority of the DSDL can be further highlighted.

Random errors impact the stability of predictions across the various methods. In this study, the assessment of predictive capability is conducted using the EPT from the equation (19), which is determined by the tolerance error magnitude δ. The subjective setting of the tolerance error may influence the comparison of prediction outcomes between different methods. To further investigate the stability of the DSDL model under random errors, we compare the predictive capabilities of the DSDL model with other contemporary deep learning methods across different tolerance errors. In the three chaotic systems with a relative error magnitude $\varepsilon = 0.01\%$, the EPTs obtained using different methods vary with the tolerance error magnitude (figures 4(a)–(c)). Both the mean EPTs and error bars are derived from 100 different training and testing sets for each chaotic system. As the tolerance error decreases, the EPT of each method declines. Nevertheless, despite the varying tolerance errors, the DSDL model consistently demonstrates leading predictive ability. This indicates that the DSDL predictive model maintains stability and accuracy in the presence of random errors, even under stringent assessment constraints. Similar results are observed when the random errors increasing to $\varepsilon = 1\%$ across the three chaotic systems (figures 4(d)–(f)).

3.2. Different impact of random errors on the DSDL and other contemporary deep learning methods

To thoroughly investigate the impact of random errors on the predictive models, it is essential to consider how the predictive capabilities of various methods change with different magnitudes of random errors. In the analysis of three chaotic systems, we evaluate the predictive capabilities of the DSDL model against other contemporary deep learning methods under varying magnitudes of random errors across 100 different training and test sets (figures 5(a)–(c)). The meaning of the relative errors at the horizontal axis and experimental conditions are similar to the previous description for the figure 3. Overall, a marked deterioration is observed in the predictive accuracy of all methods as the random errors increase. With a

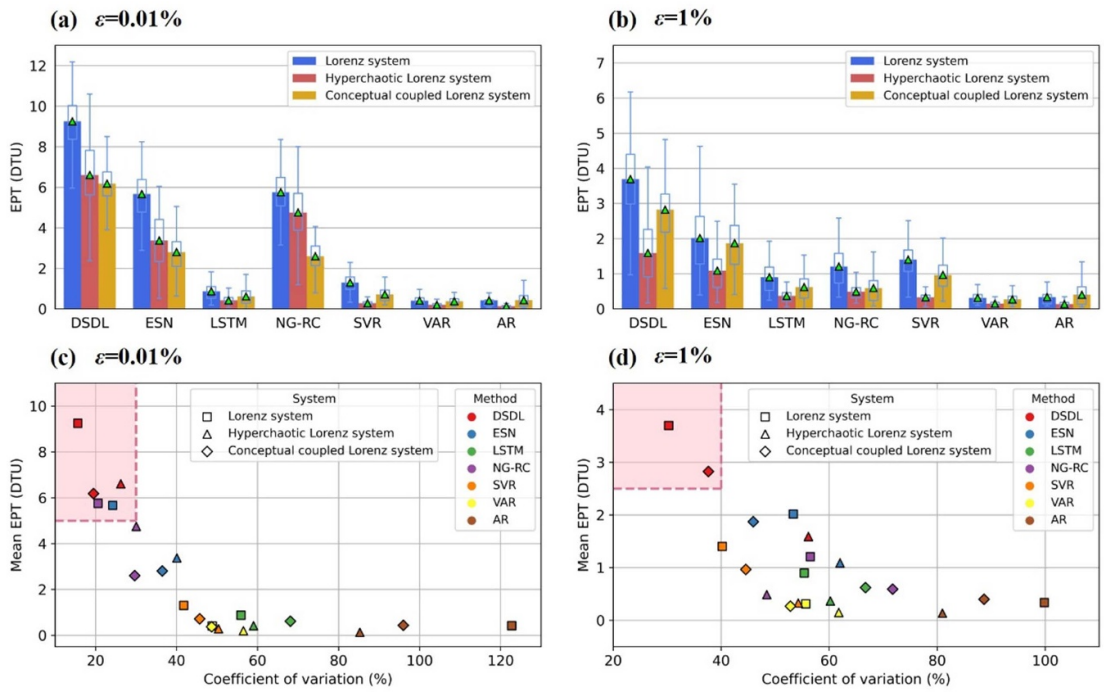

Figure 3. Comparisons between the DSDL and other existing machine learning methods in three different chaotic dynamical systems with random errors. (a) The EPTs (DTU) of different methods for three dynamical systems. Both the bars and light green triangles in the light blue boxes represent the mean. The upper and lower boundaries of the light blue boxes represent the upper and lower quartiles, respectively. The boundaries of the upper and lower whiskers represent the maxima and minima, respectively. The EPTs are obtained by 100 different training and prediction test sets with a relative error magnitude $\varepsilon = 0.01\%$. (b) Same as (a), but for the relative error magnitude $\varepsilon = 1\%$. (c) The mean EPTs (DTU) versus the coefficients of variation (%) for the systems using different methods. The dots in the pink shadow represent the superior predictive capabilities among the methods. The higher the mean EPT and the lower the coefficient of variation, the better the method performs. The training/prediction test sets are the same as the ones in (a). (d) Same as (c), but for the relative error magnitude $\varepsilon = 1\%$.

Table 1. Comparison of computational efficiency among different methods for the three chaotic systems with a relative error magnitude $\varepsilon = 0.01\%$ during training period. CPU time (CPU, units: s) and memory usage (MU, units: MB) are measured for training a predictive model.

Methods	DSDL		ESN		LSTM		NG-RC		LANDO	
Hyperparameter	No		Yes		Yes		Yes		Yes	
Metrics	CPU	MU	CPU	MU	CPU	MU	CPU	MU	CPU	MU
Lorenz	5.59	32.37	621.15	309.82	13.18	45.28	0.02	1.43	0.89	0.98
Hyperchaotic	35.39	650.41	660.64	305.58	11.54	43.33	0.03	2.40	7.37	1.56
O–A coupled	236.98	1287.13	690.96	310.57	12.98	46.36	0.03	4.82	66.23	1.60

Table 2. Comparison of computational efficiency among different methods for the three chaotic systems with a relative error magnitude $\varepsilon = 0.01\%$ during prediction period. CPU time (CPU, units: s) and memory usage (MU, units: MB) are measured for performing a single prediction.

Methods	DSDL		ESN		LSTM		NG-RC		RK4		LANDO	
Metrics	CPU	MU	CPU	MU	CPU	MU	CPU	MU	CPU	MU	CPU	MU
Lorenz	0.51	0.28	380.53	0.14	50.10	11.26	0.06	0.14	0.18	2.27	0.23	0.01
Hyperchaotic	1.10	0.37	331.58	0.15	51.65	12.04	0.18	0.20	0.20	4.57	0.21	0.03
O–A coupled	2.12	0.97	340.45	0.15	75.18	13.39	0.19	0.20	0.29	7.86	0.28	0.01

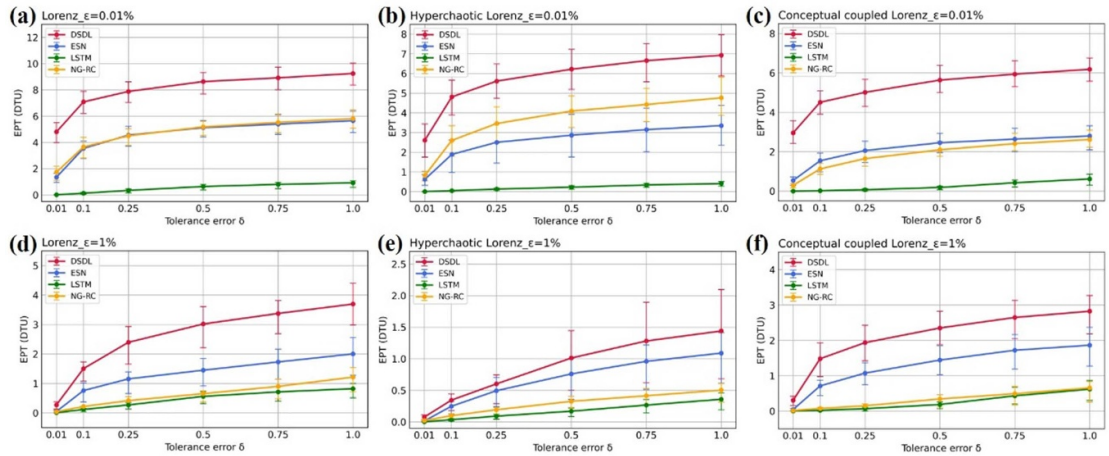

Figure 4. Predictive capabilities under different tolerance error comparing the DSDL and other existing deep learning methods in three different chaotic dynamical systems with random errors. (a) The EPTs (DTU) of different methods versus the tolerance error for the Lorenz system. Both the mean EPTs and error bars are obtained by 100 different training and prediction test sets with a relative error magnitude $\varepsilon = 0.01\%$. The upper and lower boundaries of the error bars represent the upper and lower quartiles, respectively. (b) Same as (a), but for the hyperchaotic Lorenz system. (c) Same as (a), but for the conceptual ocean–atmosphere coupled Lorenz system. (d), (e), (f) Same as (a), (b), (c) respectively, but for the relative error magnitude $\varepsilon = 1\%$.

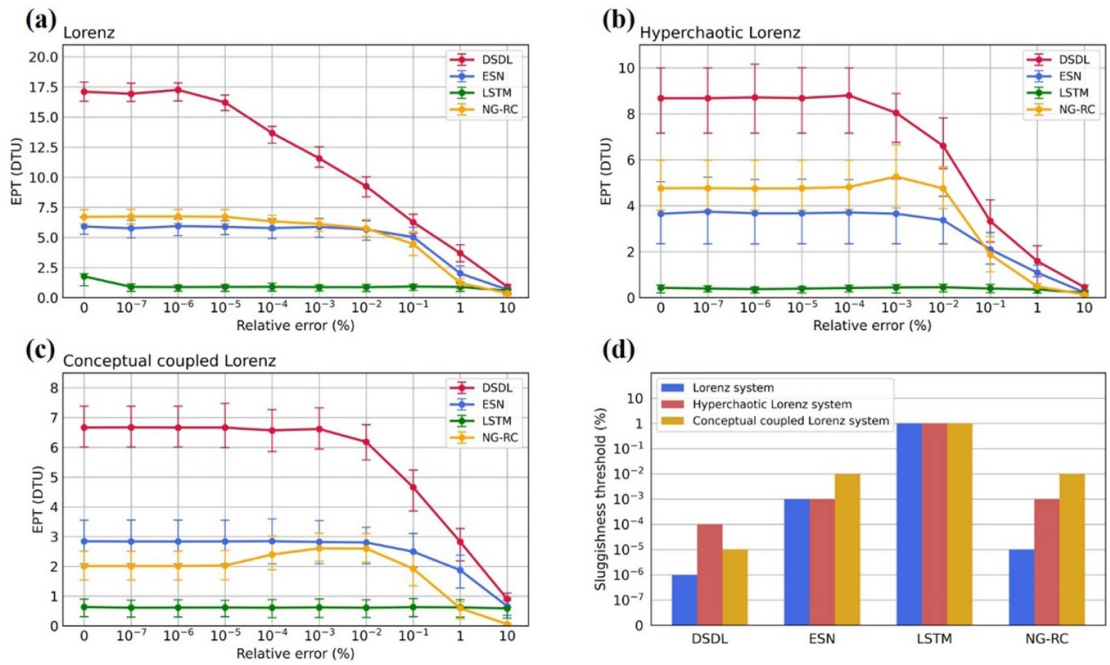

Figure 5. Impact of random errors on predictive capabilities comparing the DSDL and other existing deep learning methods in three chaotic dynamical systems. (a) The EPTs (DTU) of different methods versus the random error magnitude for the Lorenz system. Both the mean EPTs and error bars are obtained by 100 different training and prediction test sets. The upper and lower boundaries of the error bars represent the upper and lower quartiles, respectively. (b) Same as (a), but for the hyperchaotic Lorenz system. (c) Same as (a), but for the conceptual ocean–atmosphere coupled Lorenz system. (d) The sluggishness threshold (%) of different methods in the three chaotic dynamical systems.

relative error magnitude $\varepsilon = 10\%$, most methods experience a significant loss of predictive capabilities. From a comparative perspective, the DSDL model consistently demonstrates enhanced predictive performance across the various magnitudes of random errors. As the magnitude of random errors decreases, the predictive capability of the DSDL model increases rapidly, stabilizing around a high prediction level. This result suggests that reducing random errors enhances observational precision, thereby augmenting the dynamic information available to the DSDL model. Increased observational information that closely reflects the true system state strengthens the DSDL's ability to reconstruct the dynamic characteristics of the primitive system, resulting in superior predictive outcomes compared to other deep learning approaches that lack a dynamic modeling framework.

As the magnitude of random errors decreases, the predictive performance of each method initially improves, and then stabilizes (figures 5(a)–(c)). However, when each method reaches the stable predictive performance, the corresponding relative error magnitudes are different. For instance, introducing random errors in a relative error magnitude $\varepsilon = 10^{-3}\%$ for the Lorenz system, the DSDL method continues to demonstrate a significant enhancement in predictive performance compared to other methods that have nearly stabilized (figure 5(a)). This indicates that the DSDL method possesses the superior capability in recognizing the true data trajectory within the random error band, allowing it to extract more information and, consequently, make more accurate predictions. To systematically illustrate a method's capacity for recognizing the true data trajectory within the random error band, we introduce the metric of the sluggishness threshold ε_s. For a predictive method M, the sluggishness threshold ε_s represents the critical error magnitude at which the predictive performance for the target chaotic system tends to stabilize. For $\varepsilon \geqslant \varepsilon_s$, the EPT of method M significantly increases with decreasing random errors ε, expressed as $\frac{\mathrm{dEPT}_M}{\mathrm{d}\varepsilon} < 0$, indicating improved predictive capability. Conversely, for $0 \leqslant \varepsilon < \varepsilon_s$, the decrease in random errors ε has minimal impact on the EPT of method M, expressed as $\frac{\mathrm{dEPT}_M}{\mathrm{d}\varepsilon} \approx 0$, suggesting that the predictive capability remains nearly unchanged and the EPT stabilizes around $\mathrm{EPT}_M(\varepsilon_s)$. Overall, as the magnitude of random errors decreases to the sluggishness threshold, the method M becomes sluggish in recognizing the true data trajectory, with no significant improvement in performance. Thus, the prediction level is sluggish despite improving observational accuracy. Notably, the lower ε_s and the higher $\mathrm{EPT}(\varepsilon_s)$, the better the method performs. For the predictions of three chaotic systems, the sluggishness threshold of the DSDL is the lowest level among the contemporary deep learning methods (figure 5(d)), indicating the DSDL's outstanding capacity to recognize the true trajectory within the random error band. For example, in the predictions for the Lorenz system, the sluggishness threshold is approximately $\varepsilon = 10^{-6}\%$, with the corresponding $\mathrm{EPT}(\varepsilon_s)$ reaching 17.5 DTU (figures 5(a) and (d)). Clearly, both metrics are superior to those of the other deep learning methods. Similar conclusions can be drawn from the predictive results of the hyperchaotic Lorenz system and conceptual ocean–atmosphere coupled Lorenz system (figures 5(b)–(d)).

In terms of the theoretical source of the DSDL's robustness, we provide the analysis from three aspects. Firstly, the theoretical foundation of the DSDL method originates from the SSR theory [15, 19], especially the SSR under noise [51, 52], which is suitable for the prediction of dynamical systems with time-evolution properties and being applied to time series of noisy chaotic systems. The theoretical predictive relationship from the SSR ensures sufficient consideration about the correlation between the current and next states in a dynamical system. Secondly, chaotic dynamical systems contain intrinsic dynamic evolutionary properties. Even when the system is subjected to the perturbation of random errors, the fundamental evolutionary properties are still preserved within the observational data. Based on the theoretical framework of the SSR, the DSDL is capable of identifying and extracting the true evolutionary relationships intrinsic to the dynamical system within the error bands, resulting the DSDL can significantly mitigate the impact of noise. Thirdly, the DSDL demonstrates advantages in the process of selecting key variables. The SSR theory endows the DSDL model with transparency in its predictive relationships. Unlike the traditional black-box models, the DSDL can select variables that play a significant role in predicting the time evolution of the original dynamical system. The process of selecting key variables can, to some extent, eliminate interference variables that have little contribution to prediction, effectively limiting the accumulation and amplification of random errors raised by ineffective nonlinear variables during the iterative prediction process. Therefore, compared to other traditional data-driven methods, the DSDL is outstanding for capturing time-evolution and nonlinear properties of the chaotic systems, and the DSDL can perform robust and continuous predictions of chaotic systems directly based on observational data.

3.3. Comparisons between the DSDL and the previous dynamic methods

The DSDL, which operates within a dynamic framework using a data-driven approach, is affected by observational random errors in the training sets. Common numerical solving methods, such as the RK4, integrate backward from given initial values within inherent models. Observational errors also impact these numerical methods, which essentially influences the predictions generated by the numerical model. In solving chaotic dynamical systems, both the DSDL and numerical methods rely on the intrinsic dynamical relationships within the chaotic system. For a differential dynamical system, the expression can be given as follows:

$$\frac{\mathrm{d}X}{\mathrm{d}t} = F(X, R), \tag{20}$$

where variables $X = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \end{pmatrix}^{\mathrm{T}}$ and parameters $R = \begin{pmatrix} r_1 & r_2 & \cdots & r_p \end{pmatrix}^{\mathrm{T}}$ are defined in the differential equation, and the superscript T here denotes the transpose. The numerical solving method

calculates subsequent numerical solution as prediction based on the initial conditions and the primitive equations. The numerical method can be expressed as follows:

$$M\left(X\left(t_{P_0}\right), R\right), \tag{21}$$

where $X\left(t_{P_0}\right)$ represents the initial state in the prediction period. Generally, both observational errors and model errors may influence accuracy. In the numerical method, observational errors manifest as initial errors, while model errors take the form of parameter errors. The initial error is defined as the difference between the observed initial state and the true initial state during the prediction period:

$$\xi_i\left(t_{P_0}\right) = X_o\left(t_{P_0}\right) - X_R\left(t_{P_0}\right) = \varepsilon_0 X_R\left(t_{P_0}\right), \tag{22}$$

where ε_0 is defined as the relative error magnitude of the initial error, similar as the ε_0 expressed in equation (15).

The DSDL constructs the prediction model during the training period and continuously generates predictions starting from $X\left(t_{P_0}\right)$. The form of the DSDL can be expressed as:

$$\Phi\left(X\left(t_{T_0}\right), X\left(t_{T_1}\right), \cdots, X\left(t_{T_{m-1}}\right), X\left(t_{P_0}\right)\right), \tag{23}$$

where $X\left(t_{T_0}\right), X\left(t_{T_1}\right), \cdots, X\left(t_{T_{m-1}}\right)$ represent the training data. Compared to the numerical method, the DSDL incorporates the training period and does not require prior knowledge of the primitive equations. Thus, the DSDL method is primarily affected by observational errors. To clearly contrast the two methods, the introduction of observational errors in the DSDL is divided into two sections. The first section addresses initial errors, which are analogous to the observational errors specified in the numerical method. The second section introduces random errors throughout the entire training period, described as:

$$X_o\left(t\right) - X_R\left(t\right) = \xi_i\left(t\right), t \geqslant t_{T_0}, \tag{24}$$

where t_{T_0} is the initial time of the training period. The random errors $\xi_i\left(t\right)$ follow a normal distribution, and the ratio of the standard deviation of these errors to that of the primitive system is defined as the relative error magnitude ε, as specified in (16) and (17).

To examine the impact of observational errors on both methods, we compare the solving capabilities of the DSDL and RK4 in the Lorenz system. First, to ensure that the observational errors remain relatively consistent in our comparison, we focus solely on the introduction of initial errors in both the DSDL and RK4, comparing their performances under varying initial observational errors (figure 6(a)). As the initial error increases, the EPT of the DSDL remains close to that of the RK4. In the actual scenario, when we introduce random errors throughout the entire training period, we compare the DSDL with RK4 while maintaining an initial error (figure 6(b)). The details about experimental conditions can be described in figure 12. When we ensure that the magnitudes of observational errors in the DSDL and RK4 are equal (i.e. $\varepsilon = \varepsilon_0$), the solving capability of the DSDL is comparable to that of the RK4, and may even surpass it at certain relative error magnitude. It is important to note that the numerical method relies on a primitive model, which is not completely accurate. Consequently, model errors, which commonly manifest as parameter errors, must be considered for the numerical method. In this study, for example, the model error is defined as:

$$e_m = R' - R = \begin{pmatrix} 0, & 0, & 0.01 \end{pmatrix}^{\mathrm{T}}, \tag{25}$$

where R represents the true parameter and R' denotes the actual parameter with an error in the Lorenz system. The model error in formula (25) indicates that we set the parameter $r = 28.01$, while the other parameters σ and b remain the same as in the original equation (1) of the Lorenz system. Figure 6(c) displays the EPTs using the DSDL, RK4 with $r = 28$, and RK4 with $r = 28.01$ across different training and prediction test sets with observational errors. For a relative error magnitude $\varepsilon = 10^{-3}\%$, the mean EPT of the DSDL is slightly superior to that of the RK4 without parameter error. Additionally, the prediction results indicate that in 83 out of 100 sets, the DSDL outperforms the RK4 (figure 6(d)). The EPTs significantly decrease when the parameter is set to $r = 28.01$ in the Lorenz system (figure 6(c)), demonstrating that numerical methods are also affected by model errors. Model errors can lead to a reduction in the accuracy of the numerical method, in addition to observational errors. Clearly, an advantage of the numerical method is that it does not require a training period. In contrast, as a dynamical data-driven method, the DSDL directly reconstructs the dynamical prediction model from observations, without requiring knowledge of the original equations. Under the influence of observational errors, the DSDL may achieve comparable solution accuracy to numerical methods based on the true primitive equations while eliminating the need to account for potential

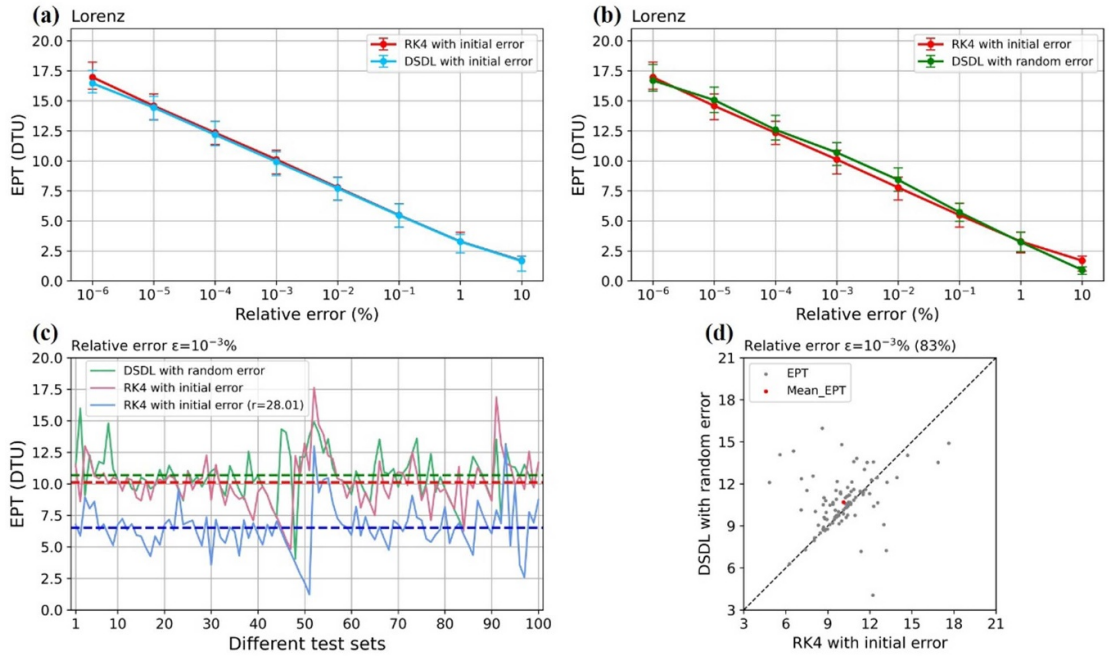

Figure 6. Impact of random errors on solving capabilities comparing the DSDL and fourth-order Runge–Kutta scheme (RK4) in the Lorenz system. (a) The EPTs (DTU) of the two methods as functions of the magnitude of initial errors. Both the mean EPTs and error bars are obtained by 100 different training and prediction test sets. The upper and lower boundaries of the error bars represent the upper and lower quartiles, respectively. (b) Same as (a), but for adding random errors in training sets of the DSDL model. (c) The EPTs (DTU) of the two methods RK4 and DSDL in different training and prediction test sets. The training/prediction test sets are the same as the ones in (b) with a relative error magnitude $\varepsilon = 10^{-3}\%$. The green, red and blue solid lines represent the EPTs of the DSDL, RK4 with $r = 28$ (true parameter) and RK4 with $r = 28.01$, respectively, and the horizontal dashed lines represent the mean EPTs of each method. (d) Scatter plot of the EPTs (DTU) of the DSDL and RK4. The training/prediction test sets are the same as the ones in (b) with the relative error magnitude $\varepsilon = 10^{-3}\%$. The grey dots represent the 100 pairs of EPTs, and the red dot represents the pair of mean EPTs. The black dashed line is the diagonal line. The value 83% represents 83 out of 100 sets where the DSDL is superior to the RK4.

model errors that arise from an incomplete understanding of the actual dynamical system. Similar results can be observed in the hyperchaotic Lorenz system and conceptual ocean–atmosphere coupled Lorenz system (shown in figures 13 and 14).

Comparing the DSDL with the RK4 as table 2 shown, the computational costs are comparable, even the memory usage of the DSDL during prediction period is lower than that of the RK4. Additionally, in real-world scenarios, known equations may contain estimation errors, leading to a notable increase in prediction errors for the RK4. In the DSDL, the internal dynamical relationships directly extracted from observational data may better approximate the true state of the system than the known equations. Therefore, through comparative analysis between the two methods, the lower computational cost and the higher predictive accuracy of prediction period imply superior predictive performance of the DSDL model.

Except the numerical solving method RK4, we compare the DSDL with a contemporary dynamic method, namely dynamic mode decomposition (DMD). The DMD was originally proposed as a data-driven approach that considers the evolutionary properties of dynamic systems, extracting dynamic information from time series [60]. The traditional DMD faces challenges in handling nonlinear systems, often adopting to local linear approximations [61]. To directly address and complement the DMD's limitations in strongly nonlinear systems, previous research proposed the linear and nonlinear disambiguation optimization (LANDO) algorithm as an enhanced variant of the DMD [62], and the LANDO was designed as a robust method to noisy conditions. Considering the capability of the LANDO and DSDL to analyze dynamic system properties and their applicability to nonlinear systems, we compare the prediction accuracy of the DSDL and LANDO under varying magnitudes of random error, as shown in figure 7. The results demonstrate that the DSDL outperforms the LANDO in prediction of the three chaotic systems. Both methods exhibit certain robustness to noise introduction, with the DSDL demonstrating superior robustness. Furthermore, as the error magnitude decreases, the DSDL shows significant improvement in predictive capability, indicating its superior ability to identify and extract dynamic evolution information from noisy observational data.

Both the DSDL and LANDO are fundamentally data-driven approaches based on the intrinsic properties of nonlinear dynamical systems. From the technical perspective, the advantages of the DSDL stem from two aspects. Firstly, the DSDL reconstructs the global attractor based on the SSR theory, especially the SSR under

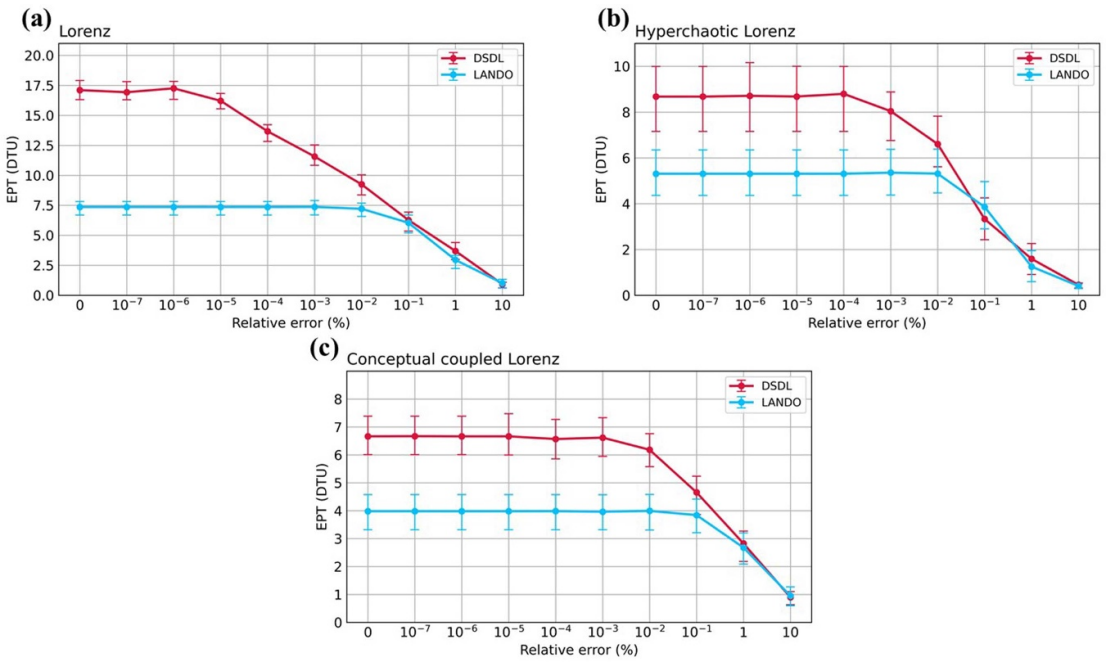

Figure 7. Same as figures 5(a) and (b), (c), but for comparing the DSDL and the LANDO in three chaotic dynamical systems.

noise, in contrast to the local linear approximation employed in the LANDO. Secondly, the DSDL introduces polynomial nonlinear terms by deeply extracting the intrinsic properties of the dynamical system to identify and introduce key variables into the predictive model. It differs from the LANDO, which adopts kernel algorithms dependent on appropriate hyperparameter optimization to introduce the nonlinear terms. Overall, the DSDL possesses the more direct and robust capability for deeply excavating the nonlinear dynamic properties of chaotic systems, resulting the superior predictive accuracy. Originating from the local linear approximation and the efficient kernel algorithms, the computational cost of the LANDO is less than that of the DSDL (tables 1 and 2). However, LANDO also necessitates the complex hyperparameter optimization process, where the computational cost is significantly increased. In future work, we will consider introducing effective techniques to improve computational efficiency of the DSDL.

4. Conclusion and discussion

In previous work, the DSDL was proposed to predict chaotic dynamical systems within a nonlinear dynamical embedding framework, demonstrating superiority over the other existing machine learning methods [41]. In this article, building on the theoretical framework of the DSDL, we newly introduce the delay embedding theorem under noisy conditions [51, 52], relatively early as the theoretical foundation for the prediction of noisy chaotic systems, thereby extending the applicability of the DSDL model. This work establishes the DSDL model under random errors and effectively predict noisy chaotic systems. Despite observational random errors present in the training sets, the DSDL model, which thoroughly considers the temporal evolution characteristics of chaotic systems, possesses the capability to identify the true system state within error bands. Therefore, the DSDL method demonstrates superior performance and robustness in predicting noisy chaotic systems.

We conclude with remarks comparing the DSDL to other predictive methods for chaotic dynamical systems. Firstly, across varying magnitudes of random errors, the DSDL outperforms contemporary machine learning methods in predicting three chaotic systems, showing robustness in handling noise during training and across diverse training and test sets. The DSDL's exceptional predictive capability reflects its ability to identify true state trajectories within the observational error bands, and to deeply extracting dynamic information from observations. From a new perspective, we introduce the concept of sluggishness threshold to describe the recognition capacity. While the DSDL needs a certain computational cost during the training period, the resulting prediction model demonstrates the advantages of high efficiency and high accuracy. Furthermore, the DSDL only requires consideration of variable introduction and faster hyperparameter tuning by using fewer parameters, which suggests the DSDL training costs are not cumbersome.

Additionally, due to the DSDL is a data-driven method grounded in dynamical systems theory, we specifically contrast the DSDL with previous representative dynamic methods, such as the numerical solving method RK4 and the LANDO, a variant of the DMD. The novel comparative numerical experimental results highlight the advantages of the DSDL's dynamical framework, demonstrating its superior robustness in predicting noisy chaotic systems. Furthermore, the DSDL's performance does not require prior knowledge of the original governing equations. The DSDL, fundamentally a data-driven approach within a dynamic framework, constructs dynamical models solely from observational data, avoiding the possible interference of model errors in the known equations.

Comparative analyses with conventional machine learning techniques and previous dynamic methods indicate that the SSR theoretical framework and the selection of key variables ensures that the DSDL is capable of identifying and extracting the true evolutionary relationships intrinsic to the dynamical system within the error bands, thereby demonstrating robust performance in predicting chaotic systems affected by random errors. This suggests the DSDL holds significant potential for reducing the impact of random errors, which remains a difficult and practical issue in the field of data-driven approaches. In real-world system predictions, the DSDL is proposed as a novel and effective tool for mitigating the impact of random errors, warranting further in-depth investigation.

Regarding the nomenclature of deep learning in the DSDL, we have considered the similarities between the DSDL and traditional deep learning, particularly in terms of multi-layer structures and nonlinearity of the layer. Traditional neural network layers are designed to handle different types of computational tasks, thereby learning complex patterns and relationships. In contrast, the layers of the multi-layer nonlinear network are used to address the task of selecting key variables from different categories, ultimately integrating all key variables that positively contribute to the chaotic system's temporal evolutionary. More importantly, the term 'deep learning' in the DSDL specifically refers to the deep exploration and extraction of the intrinsic properties of dynamical systems, with a strong emphasis on capturing the nonlinearity.

This study exclusively investigates the feasibility of the DSDL modeling prediction within deterministic chaotic systems under random errors, which naturally presents certain limitations. To effectively address prediction challenges in actual nonlinear dynamical systems influenced by observational errors, more extensive and in-depth experimental research is necessary.

Firstly, this article focuses solely on ordinary differential dynamical systems. Future research should expand to explore the prediction of partial differential dynamical systems that also incorporate random errors. For instance, actual weather and climate systems exemplify nonlinear partial differential dynamical systems where the atmospheric attractor exists [63, 64]. Enhancing predictions in these domains using the DSDL holds considerable practical significance. Our related work has demonstrated that the DSDL method can effectively predict partial differential dynamical systems, such as the Lorenz' 96 system (the system as discretized version of partial differential equation [65]) and the Kuramoto–Sivashinsky equation [66]. The accurate prediction of partial differential dynamical systems can be achieved by the DSDL model directly. However, it is important to note that a potential challenge lies in the fact that partial differential dynamical systems are often high-dimensional and complex, involving a large number of variables for modeling and processing. In future research, we aim to explore more efficient and feasible dimensionality reduction methods that can be integrated into the DSDL, thereby reducing computational costs and improving the scalability of the predictive models.

According to the tables 1 and 2, the difference in computational costs between the training and prediction periods is a critical aspect for understanding the characteristics of each methodology deeply. In the future work, we will systematically investigate the characteristics of each methodology in terms of training and prediction periods based on the findings related to computational costs.

Furthermore, the study employs the fixed time interval and training data length in the DSDL model. In real-world applications, the sampling frequency of available observational data varies. Therefore, it is crucial to consider actual data sampling to explore optimal time intervals and training lengths in the DSDL modeling for establishing effective prediction models. Lastly, the impact of random errors on prediction methods is substantial. Future research should focus on developing technical strategies to further overcome the effects of random errors, thereby improving the predictive capability of the DSDL in chaotic dynamical systems subject to observational errors.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Acknowledgment

This work was jointly supported by the National Natural Science Foundation of China (42288101), Laoshan Laboratory (LSKJ202202600), and Shandong Natural Science Foundation Project (ZR2019ZD12).

Conflict of interest

The authors declare no competing interests.

Code availability

The code used in this study is freely available upon reasonable requests.

Author contributions statement

J P Li conceived the idea. Z X Wu performed the experiments and wrote the initial draft of the manuscript. H Li and M Y Wang supplied the code and analyzed the results. N Wang and G C Liu helped perform the analysis with discussions. All authors contributed to writing the manuscript.

Appendix A

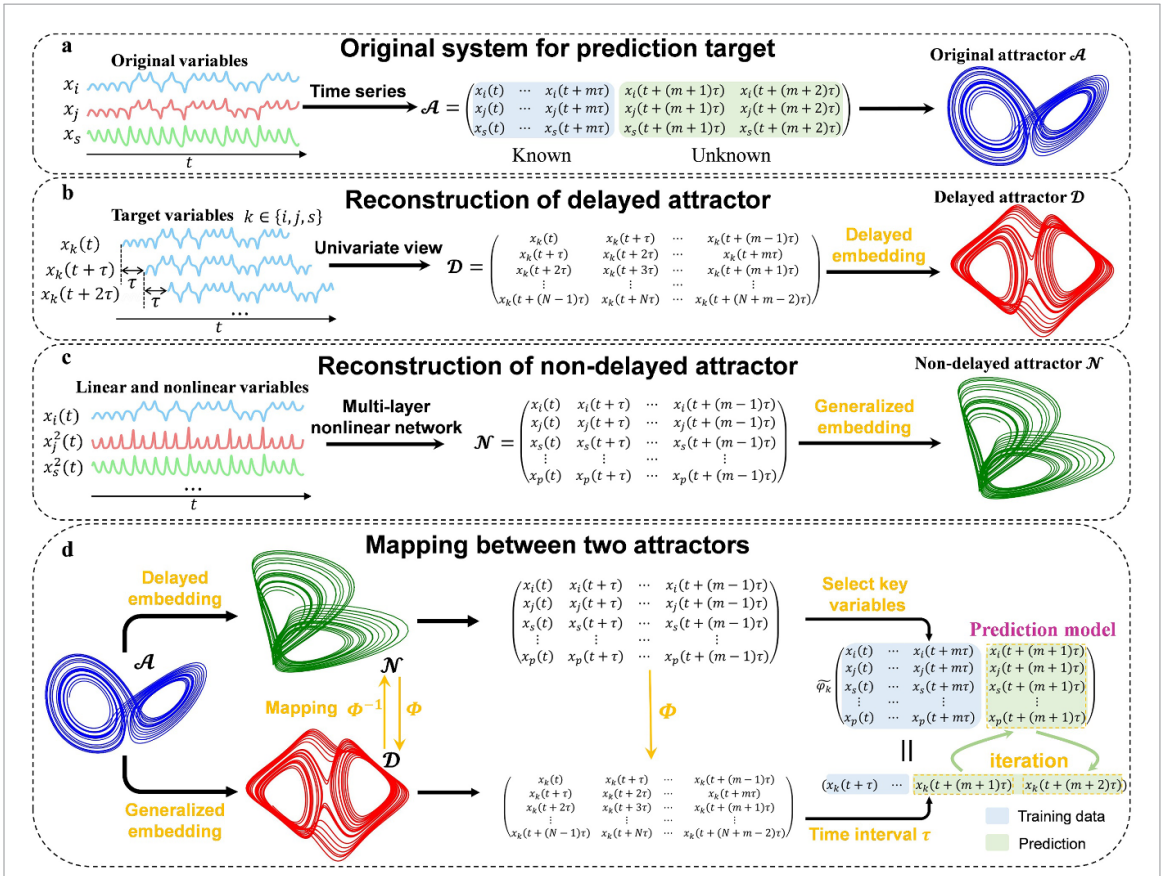

Figure 8. Sketch of DSDL framework. (a) Original system for prediction target. To predict the future unknown state of original system from known time series, each original variable (x_i, x_j, x_s) is taken as prediction target. (b) Reconstruction of delayed attractor. The original attractor can be reconstructed into the delayed attractor $\mathcal{D}$ through the delayed univariate embedding. (c) As in (b), but for the non-delayed attractor $\mathcal{N}$ through the generalized multivariate embedding, in which the multivariable is derived from multi-layer nonlinear network to deal with the nonlinearity of the prediction target system. (d) The theoretical embeddings relationship suggests a mapping between the two attractors. It is possible to fit a mapping $\widetilde{\varphi_k}$ $(k \in \{i, j, s\})$ from the multiple variables, in which we select key variables, to the future state of the target variable with a time interval τ, and carry on the iteration prediction for the original system.

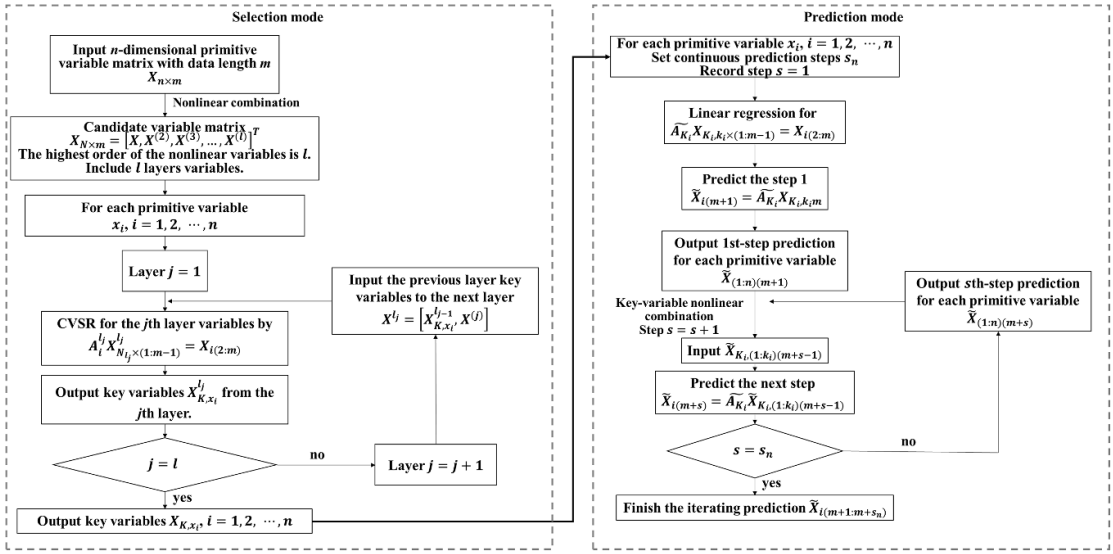

Figure 9. Flowchart of the DSDL detailed algorithm.

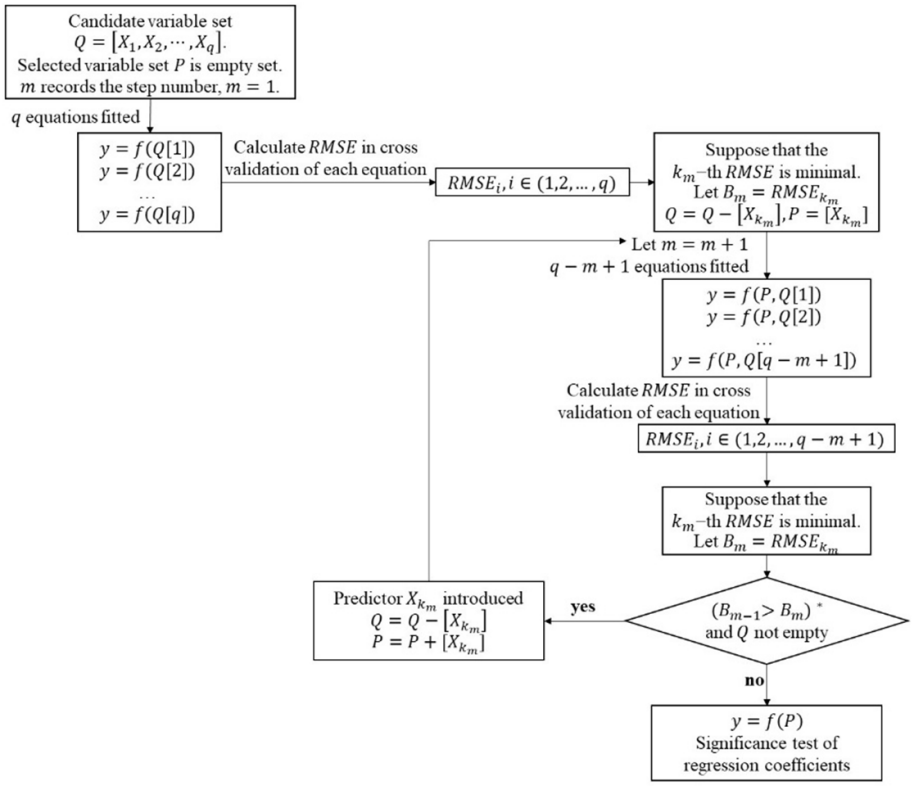

Figure 10. Flowchart of the CVSR.

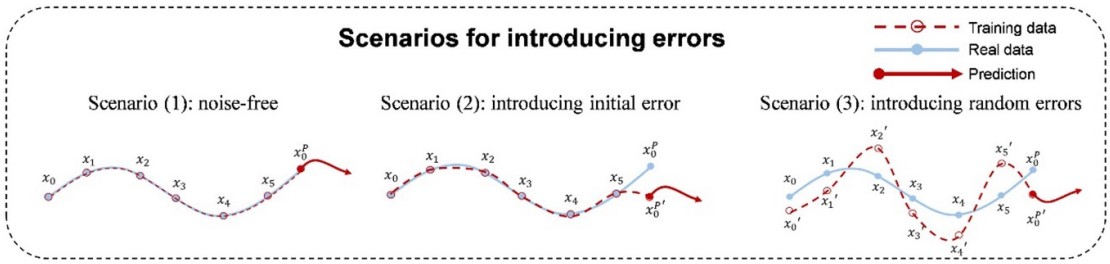

Figure 11. Sketch of error introduction scenarios.

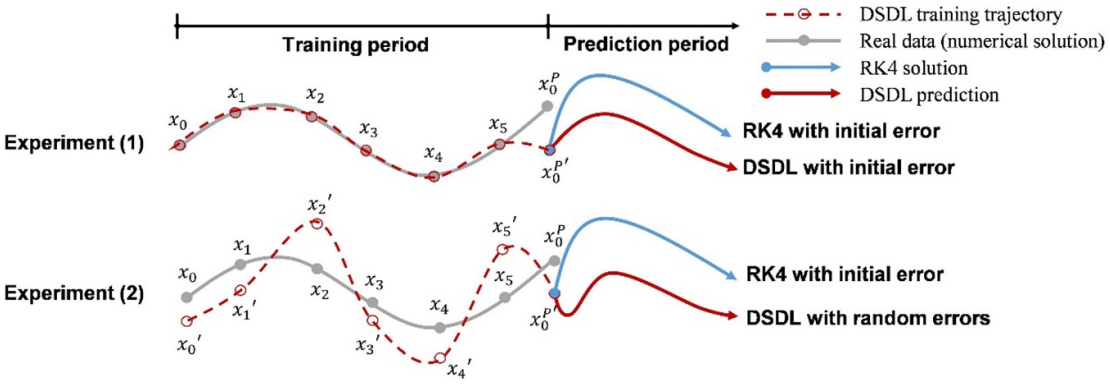

Figure 12. Diagram of the experimental error conditions for comparison between the DSDL and RK4.

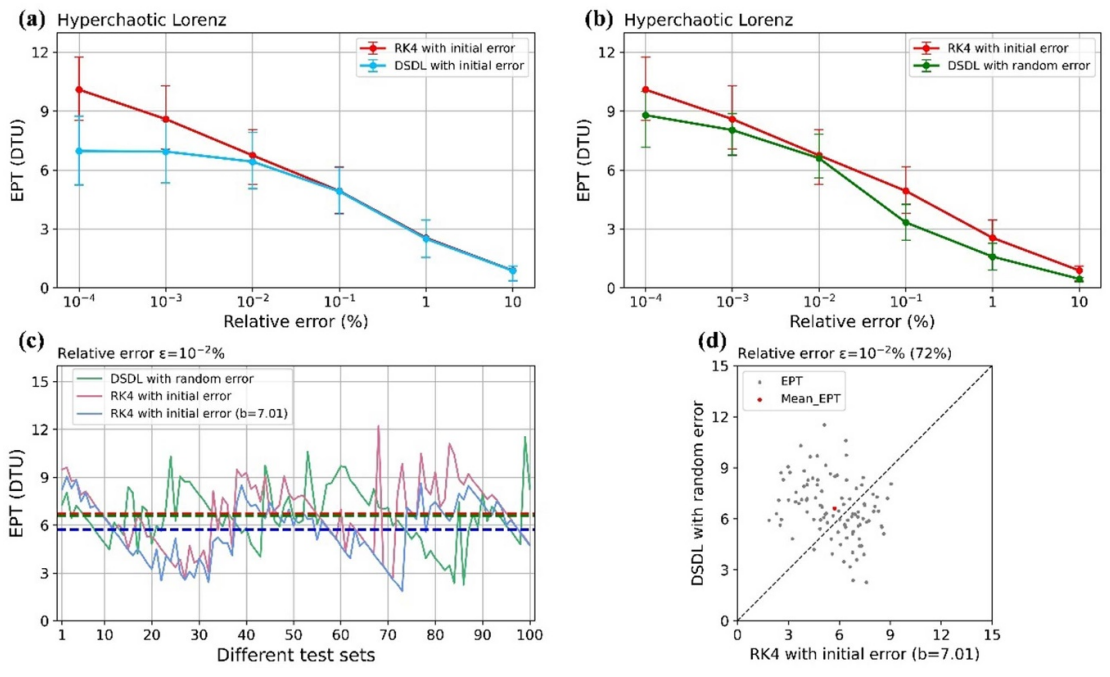

Figure 13. Same as figure 6, but for the hyperchaotic Lorenz system.

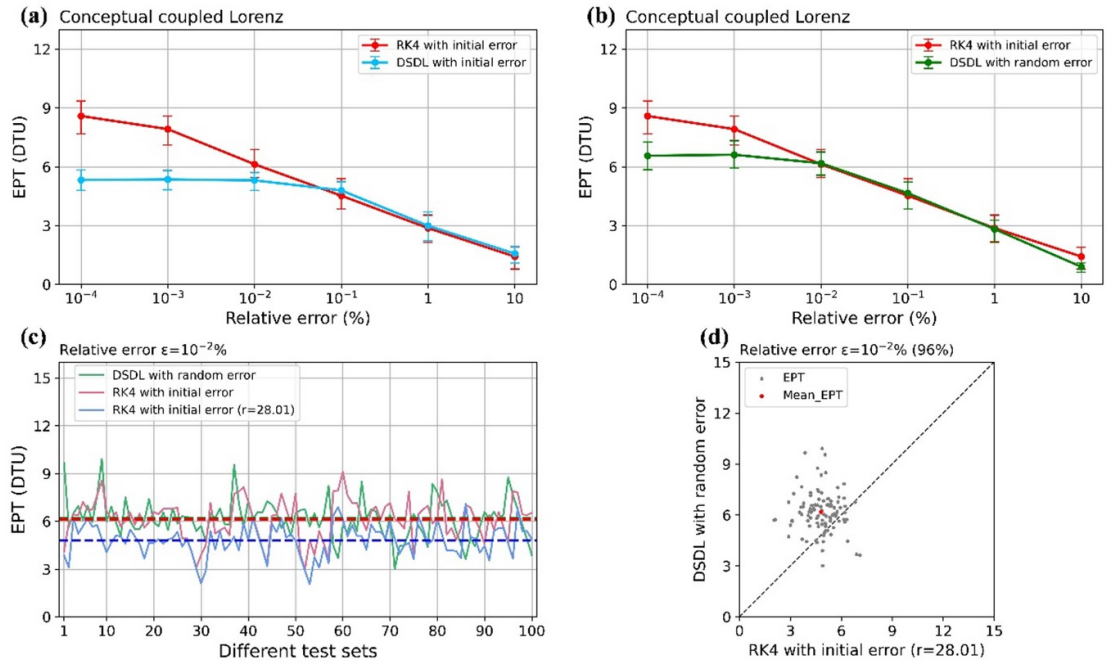

Figure 14. Same as figure 6, but for the conceptual ocean–atmosphere coupled Lorenz system.

Appendix B. Model training setup

The AR [58] uses past observations of a variable itself to predict its current and future values. Here the lag is set as 1.

The SVR [56] is the application of support vector machines in regression analysis. We set the kernel as radial basis function and the regularization parameter is 1.25.

The VAR [57] is designed to predict future values by utilizing the past values of each variable itself as well as the past values of other variables. Here the lag is set as 5.

The ESN [34] is a type of recurrent neural network characterized by a fixed and sparsely connected hidden layer, known as the reservoir, which predicts time series data through straightforward linear regression training. It does not involve backpropagation through time. Referencing [67], the codes can be accessed at https://github.com/ashesh6810/RCESN_spatio_temporal [67], and the important training parameters, such as the reservoir size and the spectral radius, are set as 5000 and 0.1.

The LSTM [30] is a form of recurrent neural network designed to model and predict sequential data by incorporating memory cells and gating mechanisms. Referencing [67], the number of epochs is set as 10; batch size is set as 72; optimization algorithm is 'adam'; the look back is set as 3. The codes can be accessed at https://github.com/ashesh6810/RCESN_spatio_temporal [67].

The NG-RC is a deep learning designed as inputting a feature vector from time-delay data to predict the next state via a linear transformation. It leverages the nonlinear vector auto-regression to build upon traditional reservoir instead recurrent neural network. Referencing [59], the codes can be accessed at https://github.com/quantinfo/ng-rc-paper-code. The ridge parameter is set as 0.014.

For [68], the code and hyperparameter settings for the (LANDO) method are available at https://github.com/PyDMD/PyDMD.

The DSDL does not involve traditional deep learning setup. The training data size for each model is uniformly set as 10 000 time points (i.e. 100 DTUs)

ORCID iDs

Jianping Li https://orcid.org/0000-0003-0625-1575
Mingyu Wang https://orcid.org/0000-0002-3246-1464
Guangcan Liu https://orcid.org/0009-0008-5736-1970

References

[1] Benincà E, Huisman J, Heerkloss R, Jöhnk K D, Branco P, Van Nes E H, Scheffer M and Ellner S P 2008 Chaos in a long-term experiment with a plankton community *Nature* **451** 822–5
[2] Nelson D M Q, Pereira A C M and de Oliveira R A 2017 Stock market's price movement prediction with LSTM neural networks *2017 Int. Joint Conf. on Neural Networks (IJCNN)* pp 1419–26
[3] Palmer T N 1993 Extended-range atmospheric prediction and the Lorenz Model *Bull. Am. Meteorol. Soc.* **74** 49–65
[4] Korn H and Faure P 2003 Is there chaos in the brain? II. Experimental evidence and related models *C. R. Biol.* **326** 787–840
[5] Gao S, Wu R, Wang X, Liu J, Li Q and Tang X 2023 EFR-CSTP: encryption for face recognition based on the chaos and semi-tensor product theory *Inform. Sci.* **621** 766–81
[6] Wang M, Fu X, Teng L, Yan X, Xia Z and Liu P 2024 A new 2D-HELS hyperchaotic map and its application on image encryption using RNA operation and dynamic confusion *Chaos Solitons Fractals* **183** 114959
[7] Gao S, Iu H H-C, Mou J, Erkan U, Liu J, Wu R and Tang X 2024 Temporal action segmentation for video encryption *Chaos Solitons Fractals* **183** 114958
[8] Ye H and Sugihara G 2016 Information leverage in interconnected ecosystems: overcoming the curse of dimensionality *Science* **353** 922–5
[9] Wang W-X, Lai Y-C and Grebogi C 2016 Data based identification and prediction of nonlinear and complex dynamical systems *Phys. Rep. Rev.* **644** 1–76
[10] Pathak J, Hunt B, Girvan M, Lu Z and Ott E 2018 Model-free prediction of large spatiotemporally chaotic systems from data: a reservoir computing approach *Phys. Rev. Lett.* **120** 024102
[11] Regazzoni F, Pagani S, Salvador M, Dede' L and Quarteroni A 2024 Learning the intrinsic dynamics of spatio-temporal processes through Latent Dynamics Networks *Nat. Commun.* **15** 1834
[12] Vlachas P R, Byeon W, Wan Z Y, Sapsis T P and Koumoutsakos P 2018 Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks *Proc. Math. Phys. Eng. Sci.* **474** 20170844
[13] Abarbanel H D I and Kennel M B 1993 Local false nearest neighbors and dynamical dimensions from observed chaotic data *Phys. Rev.* E **47** 3057–68
[14] Han M, Ren W, Xu M and Qiu T 2019 Nonuniform state space reconstruction for multivariate chaotic time series *IEEE Trans. Cybern.* **49** 1885–95
[15] Packard N H, Crutchfield J P, Farmer J D and Shaw R S 1980 Geometry from a Time Series *Phys. Rev. Lett.* **45** 712–6
[16] Takens F 1981 Detecting strange attractors in turbulence *Dynamical Systems and Turbulence, Warwick 1980* vol 898, ed D Rand and L-S Young (Springer) pp 366–81
[17] Farmer J D and Sidorowich J J 1987 Predicting chaotic time series *Phys. Rev. Lett.* **59** 845–8
[18] Sauer T, Yorke J A and Casdagli M 1991 Embedology *J. Stat. Phys.* **65** 579–616
[19] Deyle E R and Sugihara G 2011 Generalized theorems for nonlinear state space reconstruction *PLoS One* **6** e18295
[20] Gutman Y, Qiao Y and Szabó G 2018 The embedding problem in topological dynamics and Takens' theorem *Nonlinearity* **31** 597
[21] Ma H, Zhou T, Aihara K and Chen L 2014 Predicting time series from short-term high-dimensional data *Int. J. Bifurcation Chaos* **24** 1430033
[22] Ma H, Leng S, Aihara K, Lin W and Chen L 2018 Randomly distributed embedding making short-term high-dimensional data predictable *Proc. Natl Acad. Sci. USA* **115** E9994–E10002
[23] Chen P, Liu R, Aihara K and Chen L 2020 Autoreservoir computing for multistep ahead prediction based on the spatiotemporal information transformation *Nat. Commun.* **11** 4568
[24] Wu T, Gao X, An F, Sun X, An H, Su Z, Gupta S, Gao J and Kurths J 2024 Predicting multiple observations in complex systems through low-dimensional embeddings *Nat. Commun.* **15** 2242
[25] LeCun Y, Bengio Y and Hinton G 2015 Deep learning *Nature* **521** 436–44
[26] Lin Y, Liu S, Yang H and Wu H 2021 Stock trend prediction using candlestick charting and ensemble machine learning techniques with a novelty feature engineering scheme *IEEE Access* **9** 101433–46
[27] Zhang Y, Long M, Chen K, Xing L, Jin R, Jordan M I and Wang J 2023 Skilful nowcasting of extreme precipitation with NowcastNet *Nature* **619** 526–32
[28] Bi K, Xie L, Zhang H, Chen X, Gu X and Tian Q 2023 Accurate medium-range global weather forecasting with 3D neural networks *Nature* **619** 533–8
[29] Gao S, Liu J, Iu H H-C, Erkan U, Zhou S, Wu R and Tang X 2024 Development of a video encryption algorithm for critical areas using 2D extended Schaffer function map and neural networks *Appl. Math. Model.* **134** 520–37
[30] Hochreiter S and Schmidhuber J 1997 Long short-term memory *Neural Comput.* **9** 1735–80
[31] Cheng W, Wang Y, Peng Z, Ren X, Shuai Y, Zang S, Liu H, Cheng H and Wu J 2021 High-efficiency chaotic time series prediction based on time convolution neural network *Chaos Solitons Fractals* **152** 111304
[32] Meiyazhagan J, Sudharsan S and Senthilvelan M 2021 Model-free prediction of emergence of extreme events in a parametrically driven nonlinear dynamical system by deep learning *Eur. Phys. J.* B **94** 156
[33] Duan J, Zuo H, Bai Y, Chang M, Chen X, Wang W, Ma L and Chen B 2023 A multistep short-term solar radiation forecasting model using fully convolutional neural networks and chaotic aquila optimization combining WRF-Solar model results *Energy* **271** 126980
[34] Jaeger H and Haas H 2004 Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication *Science* **304** 78–80
[35] Li X, Zhu Q, Zhao C, Duan X, Zhao B, Zhang X, Ma H, Sun J and Lin W 2024 Higher-order Granger reservoir computing: simultaneously achieving scalable complex structures inference and accurate dynamics prediction *Nat. Commun.* **15** 2506
[36] Reichstein M, Camps-Valls G, Stevens B, Jung M, Denzler J, Carvalhais N and PrabhatF 2019 Deep learning and process understanding for data-driven Earth system science *Nature* **566** 195–204
[37] Wang J, Chen C, Zheng Z, Chen L and Zhou Y 2022 Predicting high-dimensional time series data with spatial, temporal and global information *Inform. Sci.* **607** 477–92
[38] Shen C *et al* 2023 Differentiable modelling to unify machine learning and physical models for geosciences *Nat. Rev. Earth Environ.* **4** 552–67
[39] Chen Z, Liu Y and Sun H 2021 Physics-informed learning of governing equations from scarce data *Nat. Commun.* **12** 6136
[40] Chen J, Shi J, He A and Fang H 2024 Data-driven solutions and parameter estimations of a family of higher-order KdV equations based on physics informed neural networks *Sci. Rep.* **14** 23874

[41] Wang M and Li J 2024 Interpretable predictions of chaotic dynamical systems using dynamical system deep learning *Sci. Rep.* **14** 3143
[42] Wang M and Li J 2024 Exploring the potential of contemporary deep learning methods in purifying polluted information *Mach. Learn : Sci Technol.* **5** 045026
[43] Lorenz E N 1963 Deterministic nonperiodic flow *J. Atmos. Sci.* **20** 130–41
[44] Sardeshmukh P D, Wang J-W A, Compo G P and Penland C 2023 Improving atmospheric models by accounting for chaotic physics *J. Clim.* **36** 5569–85
[45] López-Caraballo C H, Lazzús J A, Salfate I, Rojas P, Rivera M and Palma-Chilla L 2015 Impact of noise on a dynamical system: prediction and uncertainties from a swarm-optimized neural network *Comput. Intell. Neurosci.* **2015** 145874
[46] López-Caraballo C H, Salfate I, Lazzús J A, Rojas P, Rivera M and Palma-Chilla L 2016 Mackey-Glass noisy chaotic time series prediction by a swarm-optimized neural network *J. Phys.: Conf. Ser.* **720** 012002
[47] Sangiorgio M, Dercole F and Guariso G 2021 Forecasting of noisy chaotic systems with deep neural networks *Chaos Solitons Fractals* **153** 111570
[48] Sheng C, Zhao J, Liu Y and Wang W 2012 Prediction for noisy nonlinear time series by echo state network based on dual estimation *Neurocomputing* **82** 186–95
[49] Casdagli M, Eubank S, Farmer J D and Gibson J 1991 State space reconstruction in the presence of noise *Physica* D **51** 52–98
[50] Casdagli M 1992 Chaos and deterministic versus stochastic non-linear modeling *J. R. Stat. Soc.* B **54** 303–28
[51] Yap H L, Eftekhari A, Wakin M B and Rozell C J 2014 A first analysis of the stability of Takens' embedding *2014 IEEE Global Conf. on Signal and Information Processing (Globalsip)* pp 404–8
[52] Eftekhari A, Yap H L, Wakin M B and Rozell C J 2018 Stabilizing embedology: geometry-preserving delay-coordinate maps *Phys. Rev.* E **97** 022222
[53] Li Y X, Wallace K S and Chen G R 2005 Hyperchaos evolved from the generalized Lorenz equation *Int. J. Circuit Theory Appl.* **33** 235–51
[54] Zhang S Q 2011 A study of impacts of coupled model initial shocks and state–parameter optimization on climate predictions using a simple pycnocline prediction model *J. Clim.* **23** 6210–26
[55] Guo Y, Li J and Li Y 2012 A time-scale decomposition approach to statistically downscale summer rainfall over north China *J. Clim.* **25** 572–91
[56] Vapnik V, Golowich S E and Smola A 1996 Support vector method for function approximation, regression estimation and signal processing *Proc. 9th Int. Conf. on Neural Information Processing Systems* (MIT Press) pp 281–7
[57] Runkle D E 1987 Vector autoregressions and reality *J. Bus. Econ. Stat.* **5** 437–42
[58] Kishida K 1990 Autoregressive model analysis and decay ratio *Ann. Nucl. Energy* **17** 157–60
[59] Gauthier D J, Bollt E, Griffith A and Barbosa W A S 2021 Next generation reservoir computing *Nat. Commun.* **12** 5564
[60] Schmid P J 2010 Dynamic mode decomposition of numerical and experimental data *J. Fluid Mech.* **656** 5–28
[61] Tu J H, Rowley C W, Luchtenburg D M, Brunton S L and Kutz J N 2014 On dynamic mode decomposition: theory and applications *J. Comput. Dyn.* **1** 391–421
[62] Baddoo P J, Herrmann B, McKeon B J and Brunton S L 2022 Kernel learning for robust dynamic mode decomposition: linear and nonlinear disambiguation optimization *Proc. R. Soc.* A **478** 20210830
[63] Li J P and Chou J F 1997 Existence of the atmosphere attractor *Sci. China* D **40** 215–24
[64] Li J P and Chou J F 2003 Global analysis theory of climate system and its applications *Chin. Sci. Bull.* **48** 1034–9
[65] Basnarkov L and Kocarev L 2012 Forecast improvement in Lorenz 96 system *Nonlinear Process. Geophys.* **19** 569–75
[66] Li H, Li J, Wu Z, Wang M, Liu G, Sun R, Li R, Wang N, Song H and Zhen S 2025 Dynamics-based predictions of infinite-dimensional complex systems using Dynamical System Deep Learning method *Mach Learn : Sci Technol.* (https://doi.org/10.1088/2632-2153/adc53b) accepted
[67] Chattopadhyay A, Hassanzadeh P and Subramanian D 2020 Data-driven predictions of a multiscale Lorenz 96 chaotic system using machine-learning methods: reservoir computing, artificial neural network, and long short-term memory network *Nonlinear Process. Geophys.* **27** 373–89
[68] Ichinaga S M, Andreuzzi F, Demo N, Tezzele M, Lapo K, Rozza G, Brunton S L and Kutz J N 2024 PyDMD: a python package for robust dynamic mode decomposition *J. Mach. Learn. Res.* **25** 1–9 (available at: https://jmlr.org/papers/v25/24-0739.html)