A self-driven ESN-DSS approach for effective COVID-19 time series prediction and modelling

Weiye Wang[1,2], Qing Li[1,2] and Junsong Wang[3]

[1]School of Automation, Beijing Information Science and Technology University, Beijing, China; [2]Ministry of Education Key Laboratory of Modern Measurement and Control Technology, Beijing, China and [3]College of Big Data and Internet, Shenzhen Technology University, Shenzhen, China

Abstract

Since the outbreak of the COVID-19 epidemic, it has posed a great crisis to the health and economy of the world. The objective is to provide a simple deep-learning approach for predicting, modelling, and evaluating the time evolutions of the COVID-19 epidemic. The Dove Swarm Search (DSS) algorithm is integrated with the echo state network (ESN) to optimize the weight. The ESN-DSS model is constructed to predict the evolution of the COVID-19 time series. Specifically, the self-driven ESN-DSS is created to form a closed feedback loop by replacing the input with the output. The prediction results, which involve COVID-19 temporal evolutions of multiple countries worldwide, indicate the excellent prediction performances of our model compared with several artificial intelligence prediction methods from the literature (e.g., recurrent neural network, long short-term memory, gated recurrent units, variational auto encoder) at the same time scale. Moreover, the model parameters of the self-driven ESN-DSS are determined which acts as a significant impact on the prediction performance. As a result, the network parameters are adjusted to improve the prediction accuracy. The prediction results can be used as proposals to help governments and medical institutions formulate pertinent precautionary measures to prevent further spread. In addition, this study is not only limited to COVID-19 time series forecasting but also applicable to other nonlinear time series prediction problems.

CAMBRIDGE
UNIVERSITY PRESS

1. Introduction

Since the spread of the novel Corona Virus Pneumonia (COVID-19) epidemic in December 2019, the epidemic has triggered a serious health and economic crisis worldwide. COVID-19 exhibits high transmissibility, high mutation rates, and immune evasion properties [1]. The re-infection, recurrence, or seasonal epidemic of COVID-19 or new variant remains a substantial global public health threat. Accordingly, for the present situation, to carry out an effective prevention and control strategy, using mathematical or data-driven artificial intelligence (AI) models to analyse and predict disease spread will be of great help. Nowadays, there are many studies to make progress in COVID-19 spread analysis and prediction using mathematical or AI-based models [2].

In modelling the pandemic mathematically, the COVID-19 spread can be modelled using differential equations. Typical methods have the susceptible-infectious-recovered (SIR), the susceptible-exposed infectious recovered, and the relational improved model. In mathematical epidemic models, there are also many other types of research to forecast the temporal evolutions of COVID-19, including the non-central beta probability distribution approach, reduced-space Gaussian process regression, fractional nonlinear grey Bernoulli model, the Bayesian structural time series models, and so forth [3].

AI techniques are applied extensively in predictions of the COVID-19 epidemic due to their accuracy. In machine learning-based prediction methods, the authors used active learning models to predict the multimodal data. For predicting the COVID-19 epidemic, there are other machine learning methods, including the logistic, support vector regression (SVR), spatiotemporal autoregressive models, and so forth [4]. Furthermore, the research related to deep learning for predicting the COVID-19 time evolution has also been widely concerned. Some studies used multiple deep-learning approaches for predicting the COVID-19 epidemiological data and compared the prediction accuracy of these different methods. In addition, some studies estimate the effect of weather parameters on transmissions of COVID-19, such as temperature, rainfall, humidity, and solar irradiation, and could reflect a seasonal outbreak profile. There are some related research reviews about the forecast of the COVID-19 pandemic [5].

However, the uncertainness of COVID-19 time series evolution could lead to difficulties in mathematical modelling, and the limited availability of temporal data restricts the accuracy of prediction by using AI methods. The temporal evolution prediction of COVID-19 belongs to a sort of nonlinear time series prediction. Many researchers have shown that recurrent neural networks (RNNs) exhibit significant performances in nonlinear time series forecasts. However,

the computational cost of the RNN training process is expensive owing to the back propagation through time algorithm in the training process, and gradient vanishing or explosion problems easily happen [6]. To address these issues, some variants of RNN, such as long short-term memory (LSTM) and gated recurrent units (GRU), have been proposed and widely used to tackle the tasks of nonlinear time series forecasts. However, these models still have relatively high computational complexity. The simple methods are used to predict the evolutions of complex nonlinear time series.

Echo state network (ESN) is a highly efficient computational framework based on RNN, also known as reservoir computing, and thus it is suitable for nonlinear time series forecast. The reason why ESN is efficient is that only the output weight matrix needs to be trained by a simple linear regression algorithm, whereas other connection matrices do not need to be trained, but are assigned randomly. This simple and fast learning process can enormously reduce the computational expenses of training compared to standard RNN, LSTM, and GRU. ESN can be successful in performing computationally hard tasks and has been proven to be very effective in solving problems in many application fields, including dynamical systems modelling, chaotic time series prediction, equalizing communication channels, speech recognition and polyphonic music processing, classification tasks, meteorological forecasting, industrial plants monitoring, energy consumption and wind power generation prediction, turbofan engine multi-regime time series prediction, electricity price prediction, and intelligent stock trading system [7]. The classical SIR model is adopted to produce a set of independent synthetic time series data. These generated data to train and test a classical ESN, and then the authors use the trained ESN to forecast the spread of COVID-19. ESN based on the RNN framework which has the advantage of low computational costs is very suitable for dealing with problems of complex nonlinear time series forecasting [8].

In this study, a data self-driven ESN-based COVID-19 time series prediction model is designed [9]. Specifically, the feedback loop is closed by replacing the input with the output. In simulation experiments, we forecast the number of daily cumulative confirmed cases in nine countries. Some existing deep-learning approaches in prediction at time scale, where the results indicate that our methods can achieve better prediction performance. In addition, we analyse the effect of the data self-driven ESN model parameters on predictive accuracy [10]. By selecting the appropriate data scaling ratio, the number of neurons, the leakage rate, and the spectral radius of the connection matrix of neurons, these parameters are adaptively adjusted to improve prediction performances [11, 12]. The key contribution of the given model is given below.

Novel method

This ESN-Dove Swarm Search (DSS) approach ensembles the self-driven forecasting phase that maximizes the performances in terms of forecasting the epidemics. The self-driven ESN sustains complex temporal information through the learning of training samples to autonomously evolve.

Decision-making

Forecasting the epidemic as possible as earlier without any significant delay or any other difficulties which affects the forecasting performance when compared to several AI prediction approaches which includes the RNN, GRU's, LSTM, and variational auto encoder (VAE).

Hyperparameter tuning

The hyperparameters of the ESN algorithm are updated and optimized with the use of the DSS algorithm and thus the forecasting of COVID-19 is achieved effectively.

The paper organization of this research work is listed in Section 2 as a literature review on the forecasting of epidemics, including various existing approaches that are related to forecasting the epidemics by utilizing AI, regression, and ARIMA models, the challenges and disadvantages of the existing work and the research gap of this work. The proposed methodology of this work is explained in Section 3 including several approaches to attain effective performance. Section 4 shows the result section and the experimental analysis is analysed and determined by applying the performance evaluation, graphical representation, and comparative analysis whereas the conclusion and future work are comprised in Section 5.

2. Literature survey

MacIntyre et al. [13] implemented AI in public health to warn the epidemics at an early stage. This article concentrated on the role of AI in terms of monitoring the epidemics and the various epidemics intelligence systems that include the Health Map, Met biota Epidemics intelligence from the Open source, Epitweetr, ProMED-mail, EPIWATCH, Blue Dot, and Global Bio surveillance Portal were demonstrated. The epidemic warnings could be generated by AI technology without any human involvement. After a thorough investigation, EPIWATCH had a significant value in collecting epidemic intelligence, and the epidemic was detected and alerted at an early stage. However, AI-based epidemic warning systems were not widely used.

Uzun Ozsahin et al. [14] developed Black-Box-based Pearson Correlation approach in prediction process. The objectives of this article include the comparison of the COVID-19 cases between Greece and Israel on a weekly, a comparison of the morality rate between Greece and Israel per month, an analysis and determine the influence of vaccination and forecasting the total number of individuals who are affected by COVID-19 in Israel. The correlation analysis was utilized to achieve these objectives. After experimental analysis, the results revealed that the accuracy of the MLR was 98.2%, which is better than ANN in terms of predicting COVID-19. Meanwhile, this study failed to investigate the regression, ensemble, and optimization models that were necessary to improve the overall performance.

Arora et al. [15] evolved a concept to predict and forecast COVID-19 using the Regression as well as ARIMA models. The data were taken from Johns Hopkins University for 5 months in 2020. The Root Mean Squared Logarithmic Error (RMSLE) was employed to estimate the errors of a proposed model. After the experimental analysis, it was revealed that the RMSLE of the S-ARIMA model was 1.83, and the result of the S-ARIMA approach was superior when compared to the regression models. On the other hand, the computational process was complex.

Hasan et al. [16] presented data-driven modelling as well as predicting the COVID-19 outbreak for the policymaking of the public. In this article, the data-driven method was proposed for modelling and predicting COVID-19 and it was used by the decision-makers in handling and controlling the outbreak via non-pharmaceutical interventions. For the discrete-time stochastic augmented compartmental approach, an extended Kalman filter was employed in which the change in time of the reproduction

number was determined. The active recovered, and the confirmed cases were considered as the input. After a thorough analysis, it was revealed that the proposed approach could be utilized to develop a short to medium forecast in which COVID-19 was predicted. Meanwhile, the forecasting results were not accurate.

Saqib et al. [17] developed a concept to forecast the spread of COVID-19 progression using the hybrid polynomial Bayesian ridge regression model. The hybrid polynomial Bayesian ridge regression was utilized to formulate the model with n-degree polynomial. After the experimental analysis, it was revealed that the proposed model was better than all other approaches in terms of root mean square error (RMSE). Meanwhile, the significant factors, such as social distancing, stability of the healthcare systems, and awareness of the people were not analysed in this article.

Dash et al. [18] implemented a concept to forecast the COVID-19 outbreak using the deep learning method as SVR. The principles of the support vector machine were utilized by the SVR. Here, the LSTM and SVR approaches were utilized to determine the pandemic behaviours. The simulation outcome revealed that the rate of forecasting the pandemic behaviour by the LSTM approach was 95.5%, which was superior when compared to all other existing approaches. Meanwhile, a large number of training data were required for effective performance.

Namasudra et al. [19] demonstrated a concept to predict COVID-19 cases using the nonlinear neural network-based forecasting model. A novel nonlinear autoregressive (NAR) neural network time series (NAR-NNTS) was proposed to forecast the COVID-19 cases. The training methods which include the Levenberg–Marquardt, scaled conjugate gradient, and Bayesian regularization were employed to train the NAR-NNTS model. After the experimental analysis, it was revealed that proposed NAR-NNTS model was superior to all other approaches in predicting the data of COVID-19. Meanwhile, a large time was taken to detect COVID-19 cases.

Ketu et al. [20] implemented an enhanced Gaussian process regression-based forecasting model for COVID-19 outbreak. The multi-task Gaussian process (MTGP) regression model with enhanced predictions of novel COVID-19 outbreak was proposed in this article. The MTGP regression model was employed to forecast the outbreak of COVID-19 all over the world and it helps to plan the preventive measures to minimize the spread of the disease. After the experimental analysis, the results of the proposed model were outperformed when compared to all other approaches in terms of detecting its correctness as well as suitability. Meanwhile, this article failed to investigate the significant parameters, such as the distribution of age, population density, and the virulence of the virus which were used to improve the prediction accuracy.

Research gap

Despite the various advantages obtained from the above-mentioned literature in terms of forecasting epidemics, there are some gaps in the existing literature, which are listed below.

Absence of AI-based epidemic warning systems: The existing literature does not include the warning systems that integrate with AI, which is necessary to early warning of the epidemics. This gap is addressed in the proposed model by developing an AI-based epidemic warning system to forecast the infectious disease earlier.

Complexity in the computational process: Due to the high processing time and the availability of computational resources, the computational process is complex. This gap is addressed in the proposed model by providing efficient resources and lowering the processing time.

A large number of training data are required: The existing literature requires a large number of training data to forecast epidemics like COVID-19. The proposed model addresses this gap by predicting the epidemics with a limited number of training data.

3. Proposed methodology

The workflow of the proposed method is illustrated in Figure 1. First, the figure depicts that the data of COVID-19 affected cases. The collected datasets are subjected to a data preprocessing phase and the unwanted data are eliminated by using the minimum–maximum approach. The data self-driven ESN model is constructed for forecasting the evolution of the COVID-19 time series and then the DSS algorithm is integrated with the ESN to optimize the weight. Hence, the predicted output is determined. Finally, the performance evaluation is carried out to evaluate the accuracy of the proposed model in terms of forecasting COVID-19. Basically, the ESN-DSS model is proposed to perform the prediction of COVID-19 based on the extracted relevant factors. The input data are generated in COVID-19 time series based on the current internal state and weight. A closed feedback loop is generated that replaced the original input in the prediction process, the output of the ESN is employed with future values, and the variables are determined as input. In the feedback loop, the proposed method iteratively makes predictions for successive time steps by using its own previous predictions as input.
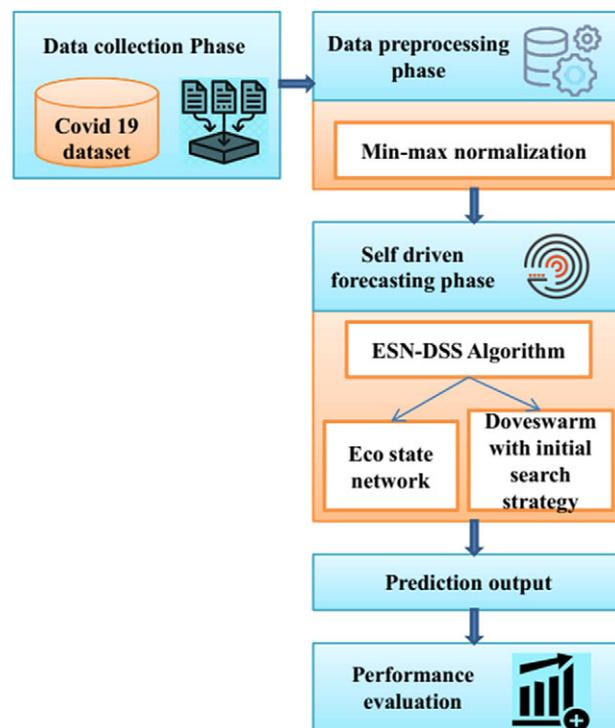

Figure 1. Workflow of the proposed method.

The feedback loop frequently updated the prediction outcome based on the output that generated self-contained and self-adaptive system. The complex temporal dependencies are formed in the closed-loop architecture leads to form more accurate and better COVID-19 prediction.

In COVID-19 prediction, the ESN is used to validate the initial factors of different signs regarding infection rate, vaccination details, and so forth. However, in DSS algorithm, it extracts the behaviour regarding the severity level that finds an effective optimal solution based on the exploration and exploitation phases. While integrating with the ESN, the DSS optimizes the weights by validating the interior issues, and the aim is achieved by identifying the optimal configuration weight by diminishing the prediction error. Initially, the weight functions are randomly assigned in the ESN, and in training process, the DSS is employed that explores the search space of possible weight configurations. The DSS model is very efficient in the process of weight updation and by integrating with ESN the performance is evaluated.

3.1. Data collection phase

On a global scale, some institutions and research teams provide many significant open-data resources against COVID-19. In our study, COVID-19 dataset comes from the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University [21]. Daily data of cumulative confirmed cases from nine considered countries are used for modelling and forecasting. These nine countries, respectively, are the United Kingdom, Canada, Australia, Italy, China, France, Japan, Spain, and the United States. Our sample dataset contains cumulative confirmed cases samples of 881 days from 22 January 2020 to 20 June 2022, and Figure 2 depicts these data samples where the number of American confirmed cases is the largest in every countries.

3.2. Data preprocessing phase

To maintain the echo states of the reservoir, inputs of the network need to be squashed into a range of $[-1, 1]$ [22]. Consequently, the transformed time series based on the min–max normalization method is described as follows:

$$y_{scale}(t) = \frac{y(t) - \min\limits_{i=1,2,\cdots,T_{train}} y(i)}{\left(\max\limits_{i=1,2,\cdots,T_{train}} y(i) - \min\limits_{i=1,2,\cdots,T_{train}} y(i)\right) \times scale}, \quad (1)$$

where $scale$ is the scaling factor, and needs to be satisfied $scale \geq 1$? It is worth noting that data squashing is determined based on the training data, while the test data are assumed to be unknown. If $scale$ is set as $scale = 1$, because what we predict is the number of daily cumulative cases, namely $y(t+1) \geq y(t)$ for any time step t, this means that outputs of the network (namely inputs) must be greater than one in the test period. This situation may affect the prediction accuracy. Therefore, in principle, it can be inferred that $scale$ cannot be less than the growth rate of the interval of the number of forecast days to ensure that the input and output of the self-driven ESN are both $[-1, 1]$ during the training and test period. If what we predict is just at an inflection point or in a period of dramatic changes, the growth rate of future time series will be much larger than that of training data. Therefore, it is difficult to determine the value of the $scale$.
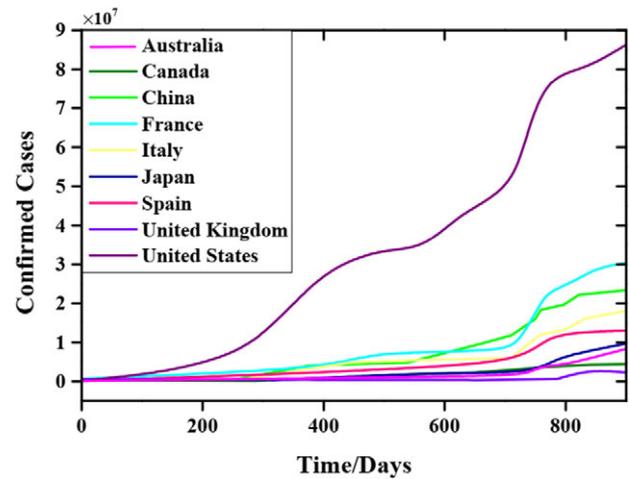

Figure 2. Actual COVID-19 time series of daily cumulative confirmed cases from 22 January 2020 to 20 June 2023, in nine different countries.

In our simulation experiments, the data scaling factor $scale$ is set as $scale = 20$ for the cumulative confirmed cases forecasting.

3.3. Self-driven forecasting phase

The objective is to provide simple deep-learning approach for modelling, predicting, and evaluating COVID-19 epidemic. The data self-driven ESN model is constructed for forecasting the evolution of the COVID-19 time series. Based on the standard ESN, the data self-driven forecast model is developed to forecast the temporal evolution of COVID-19 epidemic.

The proposed ESN-DSS model is used to predict the diseases that are influenced in covid-19 by training the insights and inform the health conditions. The time series forecasting extract the nonlinear data by associating the approaches. The weight functions are optimized that accurately identified COVID-19 time series forecasting prediction process. The parameters are tuned to enhance the flexibility of prediction process and optimized the algorithm effectively.

3.3.1. The ESN model
ESN is a highly efficient computational framework based on RNN and has a significant ability for nonlinear time series forecasting. A standard ESN architecture consists of three distinct parts, namely the input layer, dynamical reservoir, and output layer, as shown in Figure 3, where dashed arrows indicate optional connections. The input layer includes K input units, the dynamical reservoir includes N internal network units (namely reservoir neurons) and the output layer includes L output units.

For the standard ESN with leaky-integrator [23], reservoir neuron states are updated according to:

$$r(t+1) = (1-\alpha)r(t) + \alpha\mathbf{f}(\mathbf{W_{in}}u(t+1) + \mathbf{W}r(t) + \mathbf{W_{back}}y(t)),$$
$$(2)$$

where $f = (f_1, f_2, \ldots, f_N)$ is the activation function of the reservoir neuronal vector comprising scalar functions $f_1, f_2, \ldots, f_N$ typically hyperbolic tangent function (tanh). $r(t) = (r_1(t), r_2(t), \ldots, r_N(t))$ is a real-valued vector of reservoir neuron states at a time step t. Input units at the time step t are $u(t) = (u_1(t), u_2(t), \ldots, u_K(t))$
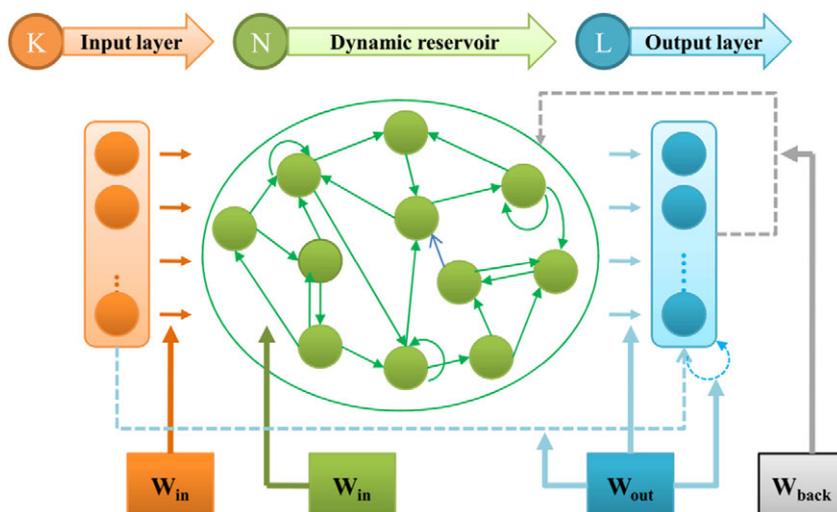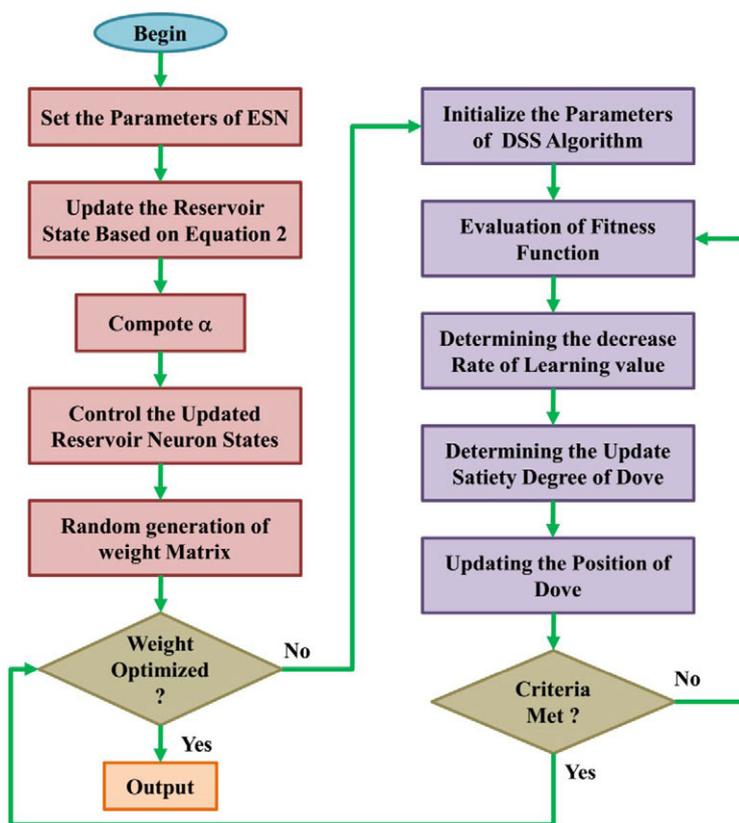
Figure 3. A standard ESN architecture.

Figure 4. Flowchart representation of the proposed ESN-DSS-based self-driven forecast model for COVID-19.

output units are $y(t) = (y_1(t), y_2(t), ..., y_K(t))$. α is the leakage rate to control the update speed of reservoir neuron states, and satisfies $0 \le \alpha \le 1$. W_{in}, W, W_{back} are the real-valued connection weight matrices whose values are randomly generated during ESN initialization. W_{in} is a $N \times K$ connection matrix that connects K inputs to N reservoir neurons, W is a $N \times N$ connection matrix that connects N reservoir neurons, and W_{back} is an $N \times L$ feedback connection matrix that projects back N reservoir

neurons from L outputs, where the feedback connection is optional but not necessary as displayed by the yellow dashed arrow in Figure 3.

Different from general RNNs, ESNs, W_{in}, W, W_{back} do not need to be trained, but are randomly determined during ESN initialization. In the subsequent training and testing process, their values remain invariant. For ESN to possess an echo state, the spectral radius of the matrix W generally needs to be satisfied
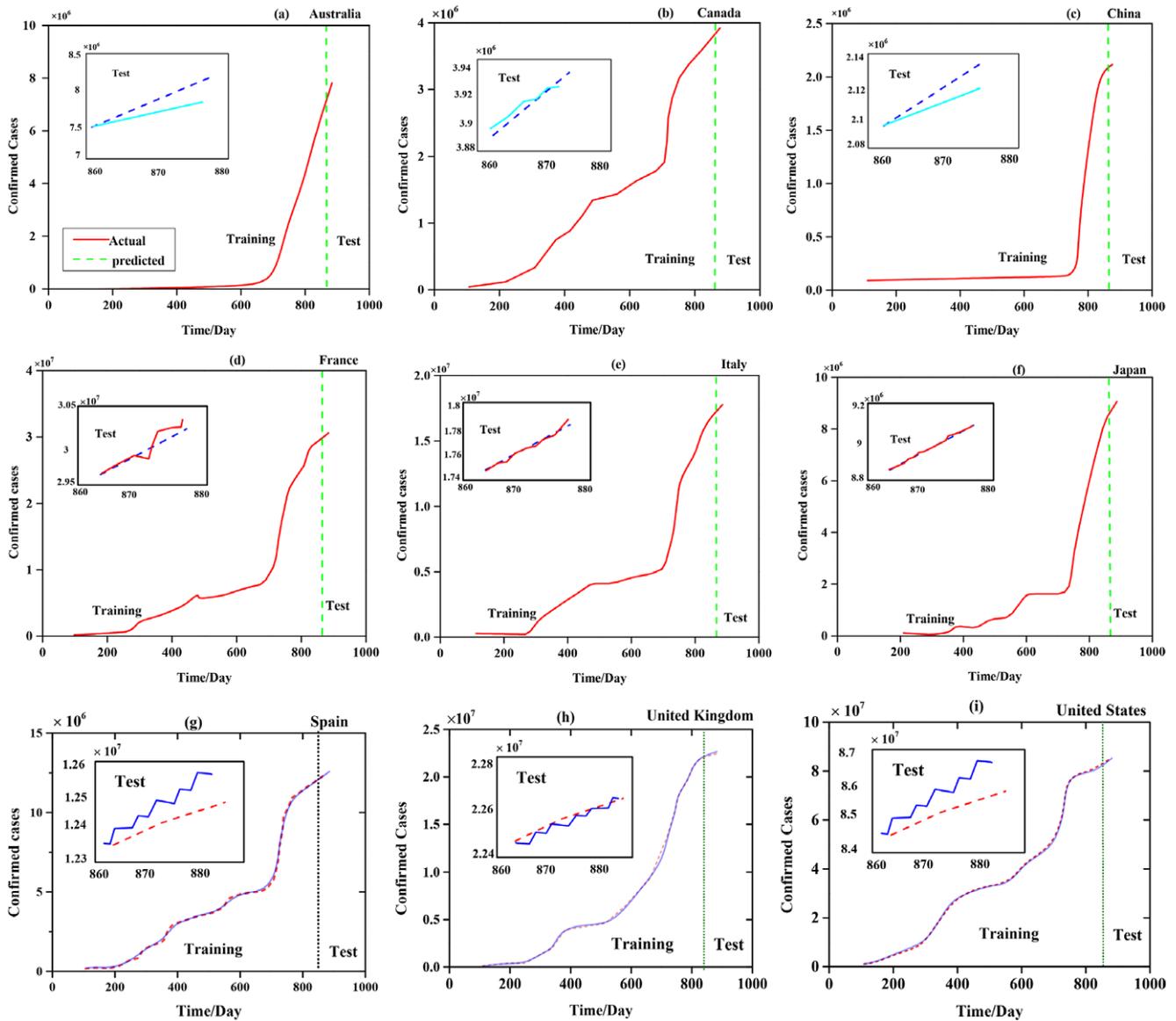
Figure 5. The comparison results of actual and predicted confirmed cases during the training and testing period for nine countries.

Table 1. Parameter settings

Parameters	Ranges
Leakage rate	$\alpha = 0.3$
Ridge regression coefficient	$\beta = 1 \times 10^{-8}$
The spectral radius of the matrix W	$\rho(W) = 0.95$
Average individual runs	50
Maximum number of iterations	100
Learning rate	0.1

$\rho(W) < 1$. However, if ESN has the weight matrix $W\rho(W) \geq 1$, the echo state characteristic may also hold [18].

The key to the ESN model is the computation of the output weight matrix W_{out}. In the training period, the dynamical

reservoir starts with an arbitrary network state $r(0)$ and then is updated according to Equation (2). If the length of the training samples is expressed T_{train} we will collect the reservoir state vectors $r(t) t = 1, 2, \ldots, T_{train}$. To dismiss the influences of the initial reservoir state, the state vectors of the initial $T_{initial}$ steps are discarded owing to the state forgetting the property of the ESN. We will obtain a $(K + N + L) \times (T_{train} - T_{initial})$ reservoir state matrix R by concatenating the input, neuron state, and output vector.

$$\mathbf{R} = \begin{bmatrix} u(T_{initial}+1) & u(T_{initial}+2) & \cdots & u(T_{train}) \\ r(T_{initial}+1) & r(T_{initial}+2) & \cdots & r(T_{train}) \\ y(T_{initial}) & y(T_{initial}+1) & \cdots & y(T_{train}-1) \end{bmatrix}. \quad (3)$$

Minimizing the mean square error (MSE) between the predicted output and the actual output W_{out} will be solved as described in Equations (4) and (5).
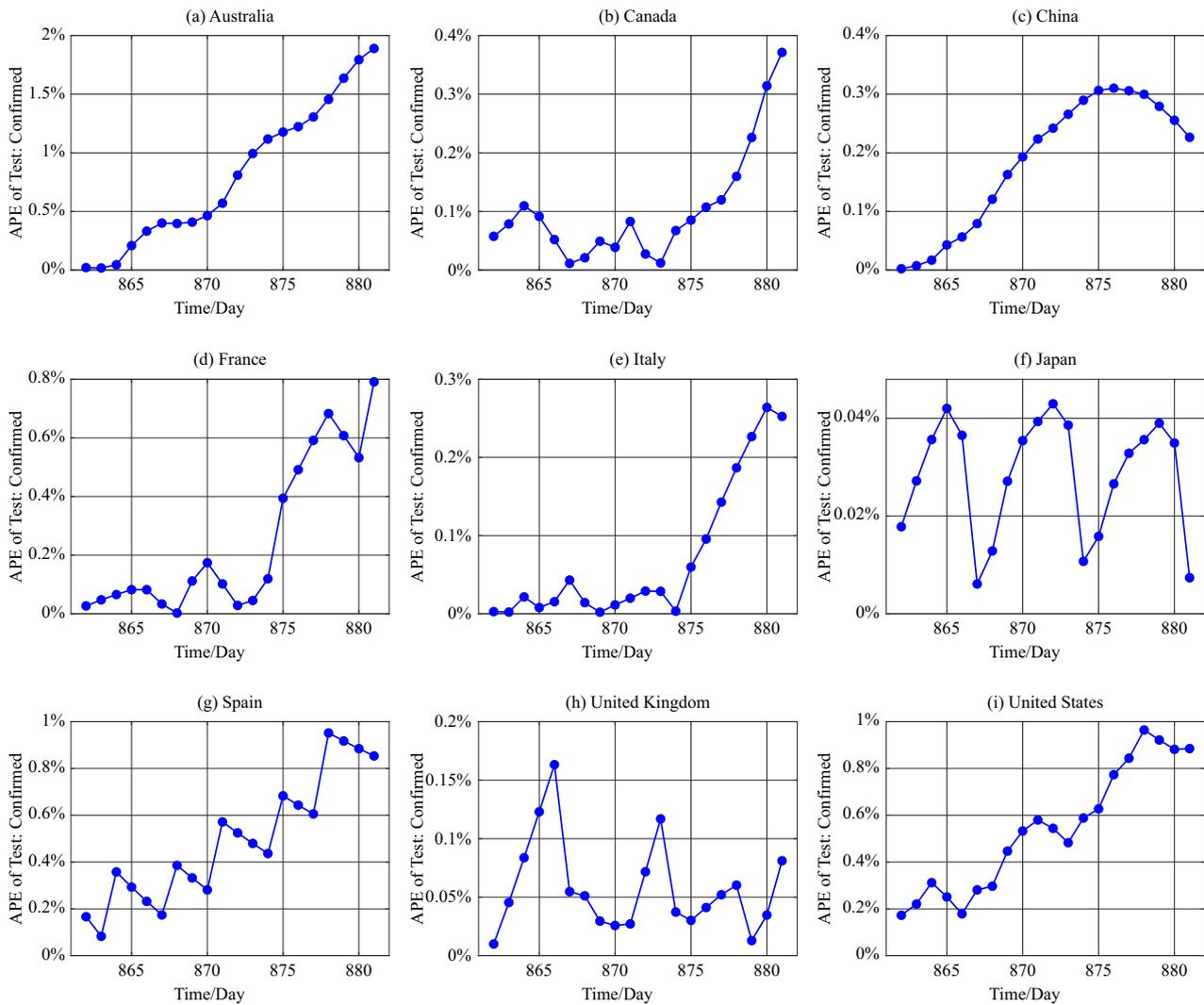
Figure 6. The changes of APE of prediction confirmed cases with time during the testing period for nine countries.

$$MSE_{train} = \left(\frac{1}{T_{train} - T_{initial}}\right) \sum_{t=T_{initial}+1}^{T_{train}} (\widehat{y}(t) - y(t))^2$$

$$= \left(\frac{1}{T_{train} - T_{initial}}\right) \sum_{t=T_{initial}+1}^{T_{train}} (\mathbf{W_{out}}(u(t), r(t), y(t-1))$$

$$- y(t))^2, \tag{4}$$

$$\mathbf{W_{out}} = \arg\min(MSE_{train}). \tag{5}$$

The way of minimizing MSE uses only the linear regression algorithms for the training, therefore, the matrix for solving W_{out} is expressed as follows:

$$\mathbf{W_{out}} = \mathbf{Y}\mathbf{R}^{\mathrm{T}}(\mathbf{R}\mathbf{R}^{\mathrm{T}} + \beta\mathbf{I})^{-1}. \tag{6}$$

$Y = [y(T_{initial}+1)y(T_{initial}+2)\ldots.y(T_{train})]$ is $L \times (T_{train} - T_{initial})$ the actual output matrix in the training period. R^T is the transpose of R. β is the ridge regression coefficient to solve over-fitting problems that may occur during training. I is $K + N + L$ the order identity matrix. The output weight matrix W_{out} can be calculated according to Equation (6).

3.3.2. DSS optimization

For various crumbs, the unsatisfied doves are shifted to the spots whereas the spots with the highest crumbs are occupied by the fed doves. To accelerate the training mechanism, two effective weight initialization approaches are employed for initializing the weight vector and constructing the feature maps [24].

Initial search strategy. In this article, to simplify the arithmetic evaluation and to avoid the boundaries of the velocity, an initial search strategy is integrated and it is given as follows:

$$c_{kl}(s+1) = c_{kl}(s) + |h_1| \left(c_{kl}^q(s)\right) + |h_2| \left(z_l^h(s) - c_{kl}(s)\right) \tag{7}$$

The valid values with every metric are in the minimal hyper-rectangle and it is represented by $[n_1, v_1], \ldots., [n_m, v_M]$. The upper and lower bounds are denoted by v_a, n_a. To squeeze the two-dimensional hyper-rectangles into the two-dimensional planes, the basic concept of initialization is employed. Hence, the two-dimensional net is used to cover the solution spaces. The weight vectors with the four neurons of the network are evaluated and it is given as follows:

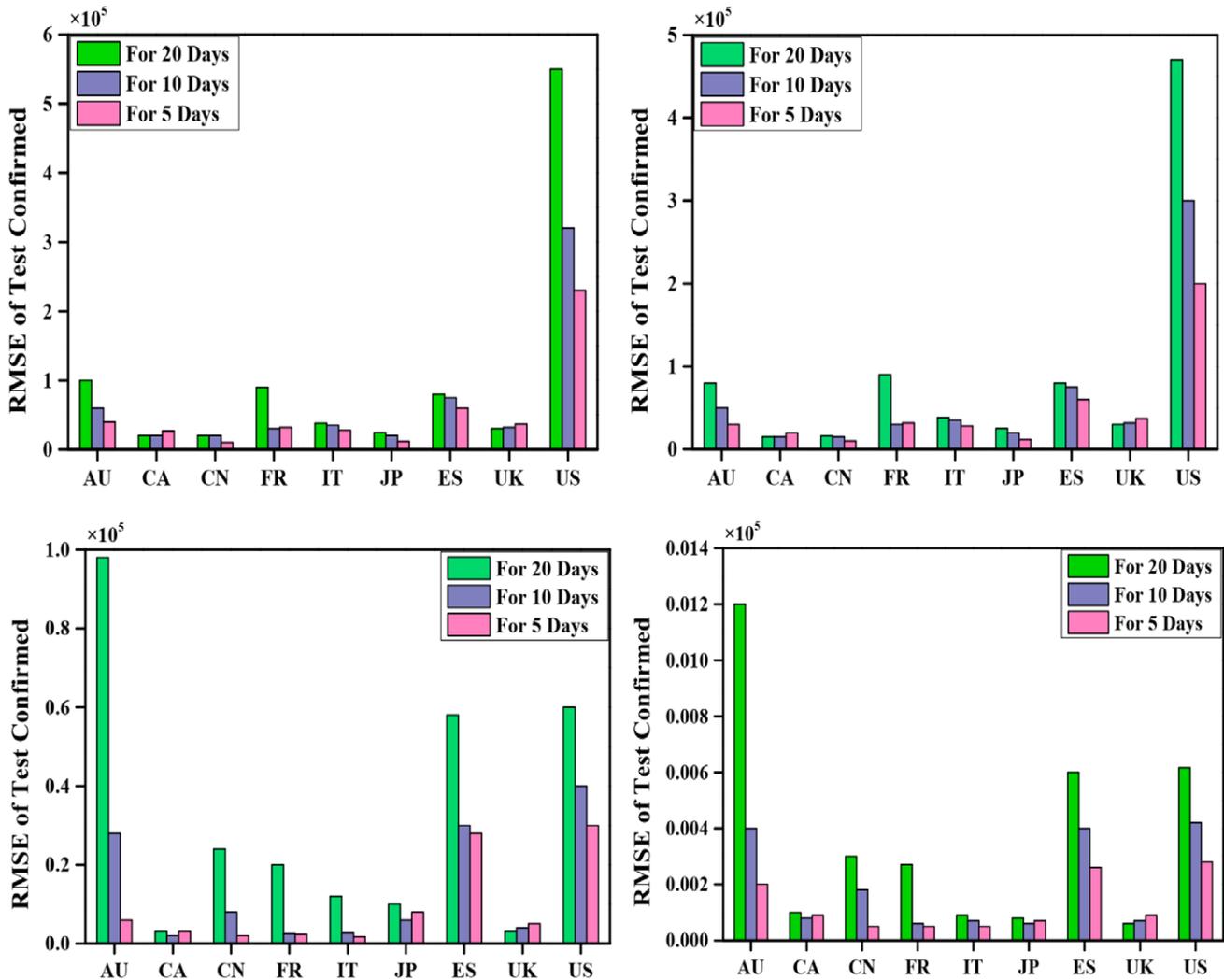$$\underline{y}_{1,1} = (n_1, n_2, \ldots., n_M)^R \tag{8}$$

Figure 7. The four testing error histograms of forecasted confirmed cases for nine countries for nine countries.

$$y_{A,Z} = (v_1, v_2, \ldots, v_M)^R \tag{9}$$

$$\underline{y_{1,Z}} = \left(n_1, n_2, \ldots, n_{\lfloor M/2 \rfloor}, v_{\lfloor M/2 \rfloor + 1}, \ldots, v_M\right)^R \tag{10}$$

$$\underline{y_{A,1}} = \left(v_1, v_2, \ldots, v_{\lfloor M/2 \rfloor}, m_{\lfloor M/2 \rfloor + 1}, \ldots, n_M\right)^R \tag{11}$$

A large number of crumbs with the use of the maximal criterion at the epoch e is evaluated and expressed as follows:

$$f_i^e = \mathrm{ARG\,MAX}\{h(y_i^e)\}, \quad for\, i = 1, \ldots, 0 \tag{12}$$

The updated satiety degree of the dove is expressed as follows:

$$S_i^e = \beta S_l^{e-1} + f^{\left(h(y_i) - h\left(y_{f_h}\right)\right)}, \quad for\, i = 1, \ldots, 0 \tag{13}$$

Most satisfying doves are selected by the high satiety degree with the use of maximal criterion

$$f_s^e = \underset{1 \le i \le O}{ARG\,MAX}\{S_i^e\}, \quad for\, i = 1, \ldots, 0 \tag{14}$$

The updated position vectors of every dove are obtained by employing the maximum criteria

$$\underline{y}_i^{e+1} = \underline{y}_i^e + \eta \alpha_i^e \left(\underline{y}_{f_s}^e - \underline{y}_i^e\right) \tag{15}$$

where

$$\alpha_i^e = \left(\frac{S_{z_s}^e - S_i^e}{S_{z_s}^e}\right)\left(1 - \frac{\left\|\underline{y}_i^e - \underline{x}_{f_s}^e\right\|}{MAX\,DISTANCE}\right) \tag{16}$$

$$MAX\,DISTANCE : \underset{1 \le i \le O}{MAX}\left\|\underline{y}_i - \underline{y}_l\right\| \tag{17}$$

From the above equation, the learning rate is denoted by η which is used in updating the position vector of the dove. The termination conditions are given as follows:

$$\left|g_{e_t}^f - U(f)\right| \le \tau_{ore} \le SET\,MAX\,EPOCH \tag{18}$$

The DSO algorithm had the complexity order and it is expressed as follows:

Table 2. The evaluation indices of testing error for confirmed cases forecasting

Country	Forecast days	RMSE	MAE	MAPE (%)	EV	RMSLE
Australia	20 days	94,979	75,966	0.9848	0.8597	0.0121
	10 days	31,186	25,629	0.3407	0.9470	0.0041
	5 days	14,853	10,364	0.1394	0.9011	0.0020
Canada	20 days	4,381	3,466	0.0886	0.8747	0.0011
	10 days	3,136	2,909	0.0746	0.9715	0.0008
	5 days	3,384	3,277	0.0841	0.9385	0.0009
China	20 days	6,118	5,233	0.2483	0.6572	0.0029
	10 days	3,169	2,452	0.1167	−0.1944	0.0015
	5 days	924	725	0.0346	0.5785	0.0004
France	20 days	77,316	56,050	0.1858	0.8877	0.0026
	10 days	16,763	14,493	0.0485	0.9674	0.0006
	5 days	13,817	13,220	0.0444	0.9851	0.0005
Italy	20 days	22,214	16,484	0.0930	0.9782	0.0013
	10 days	11,107	9,273	0.0529	0.9865	0.0006
	5 days	6,980	5,492	0.0314	0.9620	0.0004
Japan	20 days	8,956	7,114	0.0786	0.9912	0.0010
	10 days	5,116	4,423	0.0495	0.9957	0.0006
	5 days	6,478	6,294	0.0707	0.9964	0.0007
Spain	20 days	71,472	63,166	0.5053	0.7406	0.0057
	10 days	40,143	36,568	0.2943	0.7704	0.0032
	5 days	30,989	28,560	0.2304	0.6739	0.0025
United Kingdom	20 days	162,873	13,442	0.0596	0.9600	0.0007
	10 days	17,698	14,370	0.0638	0.8872	0.0008
	5 days	22,992	19,394	0.0862	−5,410.37	0.0010
United States	20 days	514,652	462,642	0.5397	0.8538	0.0060
	10 days	303,062	279,393	0.3285	0.8632	0.0036
	5 days	197,977	193,120	0.2281	0.8772	0.0023

$$Q(MM_f E) \qquad (19)$$

From the above equation, (M_f) denotes the total number of data points in the dataset, the number of data points in the dataset is denoted by $Q(MM_f E)$, e denotes the total number of epochs and the total number of doves is indicated by M.

$$f_l^e = ARGMIN\{h(y_i^e)\}, \quad for\, i = 1,\dots\dots,O \qquad (20)$$

$$S_l^e = \begin{cases} \beta S_i^{e-1} + f^{\left(h(y_i) - h\left(y_{f_h}\right)\right)}, & if\, h\left(y_{f_h}\right) \neq 0 \\ \beta S_i^{g-1} + 1, & if\, h\left(y_{f_h}\right) = 0 \end{cases} \qquad (21)$$
$$for\, i = 1,\dots\dots,O$$

To find more food, the doves are moving with a high satiety degree. Social learning is simulated for updating the position vectors $y_{f_s}^e$ through a high degree of satiety.

3.3.3. ESN-DSS-based self-driven forecast model for COVID-19
In this model, the outputs mimic the inputs. During the training period, actual training samples as teacher outputs are written back the input unit to teach the autonomous evolution of complex COVID-19 temporal information to the dynamical reservoir. In the test phase, the predictive output result feeds back the input unit in place of inputs. By replacing the input with the output, the feedback loop is closed in Figure 4.

For the self-driven ESN, outputs of the network also need to be kept in $[-1, 1]$ owing to replacing inputs with outputs. COVID-19 time series data are one-dimensional univariate.

The predicting output of ESN is described as follows:

$$\widehat{y}(t+1) = \mathbf{W_{out}}(u(t+1), r(t+1), y(t)), \qquad (22)$$

where $(u(t+1), r(t+1), y(t))$ is a concatenation vector composed of the input and neuron state vectors. W_{out} is a $L \times (K + N + L)$ matrix of connections from the inputs, neurons, and previous outputs to the current outputs, where the inputs and previous

Table 3. Comparative results for confirmed cases forecasting at the same time scale: our self-driven ESN model versus five other prediction models of deep learning

Country	Model	RMSE	MAE	MAPE (%)	EV	RMSLE
Australia	Self-driven ESN	**14**	**11**	**0.1508**	0.9232	0.0019
	RNN	39,928	397,443	547	0.279	0.0032
	GRU	295,978	293,738	4,042	0.349	0.0017
	LSTM	327,123	325,203	4,476	0.383	0.0021
	BiLSTM	335,033	333,098	4,584	0.363	0.0022
	VAE	18,732	17,186	0.236	**0.952**	**0**
China	Self-driven ESN	**63**	**52**	**0.0059**	0.7496	0.0007
	RNN	1,252,034	1,250,442	1,485	0.095	0.0002
	GRU	1,085,698	1,083,975	1,287	0.151	0.0002
	LSTM	101,482	1,013,002	1,203	0.163	0.0001
	BiLSTM	1,205,955	1,204,413	143	0.156	0.0002
	VAE	11,103	107,873	0.128	**0.843**	**0**
France	Self-driven ESN	**8,765**	**7,954**	**4.0756**	−1.11	0.0461
	RNN	1,287,786	1,279,681	6,827	0.224	0.0051
	GRU	1,204,139	1,196,438	6,383	0.311	0.0044
	LSTM	1,085,008	1,075,795	5,738	0.258	0.0036
	BiLSTM	1,168,893	1,160,923	6,193	0.308	0.0041
	VAE	3,688,083	3,522,353	188	**0.554**	**0.0004**
Italy	Self-driven ESN	**274**	**235**	**0.0995**	**0.9667**	**0.0012**
	RNN	1,070,474	1,062,061	4,519	0.201	0.0022
	GRU	113,775	1,130,957	4,813	0.314	0.0025
	LSTM	1,054,089	1,046,257	4,452	0.267	0.0021
	BiLSTM	1,041,374	1,033,467	4,398	0.269	0.0021
	VAE	1,386,225	1,385,829	5,901	0.951	0.0033
Spain	Self-driven ESN	**1,208**	**993**	**0.4084**	0.7939	0.0050
	RNN	1,683,011	167,719	6,944	0.272	0.0052
	GRU	1,795,678	1,791,683	7,419	0.467	0.006
	LSTM	1,254,449	1,247,959	5,166	0.396	0.0028
	BiLSTM	1,194,711	1,187,629	4,916	0.372	0.0026
	VAE	5,315,748	5,288,172	219	**0.891**	**0.0005**
United States	Self-driven ESN	**7,469**	**5,795**	**0.2858**	**0.9964**	0.0036
	RNN	5,227,287	5,136,497	26,373	0.208	0.0967
	GRU	4,369,108	4,240,145	21,697	0.066	0.0635
	LSTM	1,129,183	1,123,909	58,008	0	0.7589
	BiLSTM	4,330,228	4,194,141	21,451	0.024	0.0621
	VAE	4,079,244	3,976,682	204	0.993	**0.0004**

outputs are optional but not required, as shown by the red dashed arrows in Figure 3. If the red dashed arrows do not exist, W_{out} there is a $L \times N$ matrix connecting the reservoir to the outputs.

The initial population of each element is formed by

$$c_{kl}(0) = c_{\min,l} + u_{kl}(c_{mzx,k} - c_{\min,k}) \qquad (23)$$

From the above equation, $c_{mzx,k}$ denotes the upper bound, whereas $c_{\min,k}$ denotes the lower bound of the variable c_k. The

random number that is uniformly distributed at a range of $[0,1]$ is represented by u_{kl}. It is regenerated by the above equation when the element is over the domain which is allowed.

The initial rate of the learning value is fixed at 0.1, whereas the decrease rate of the learning value is evaluated and given as follows.

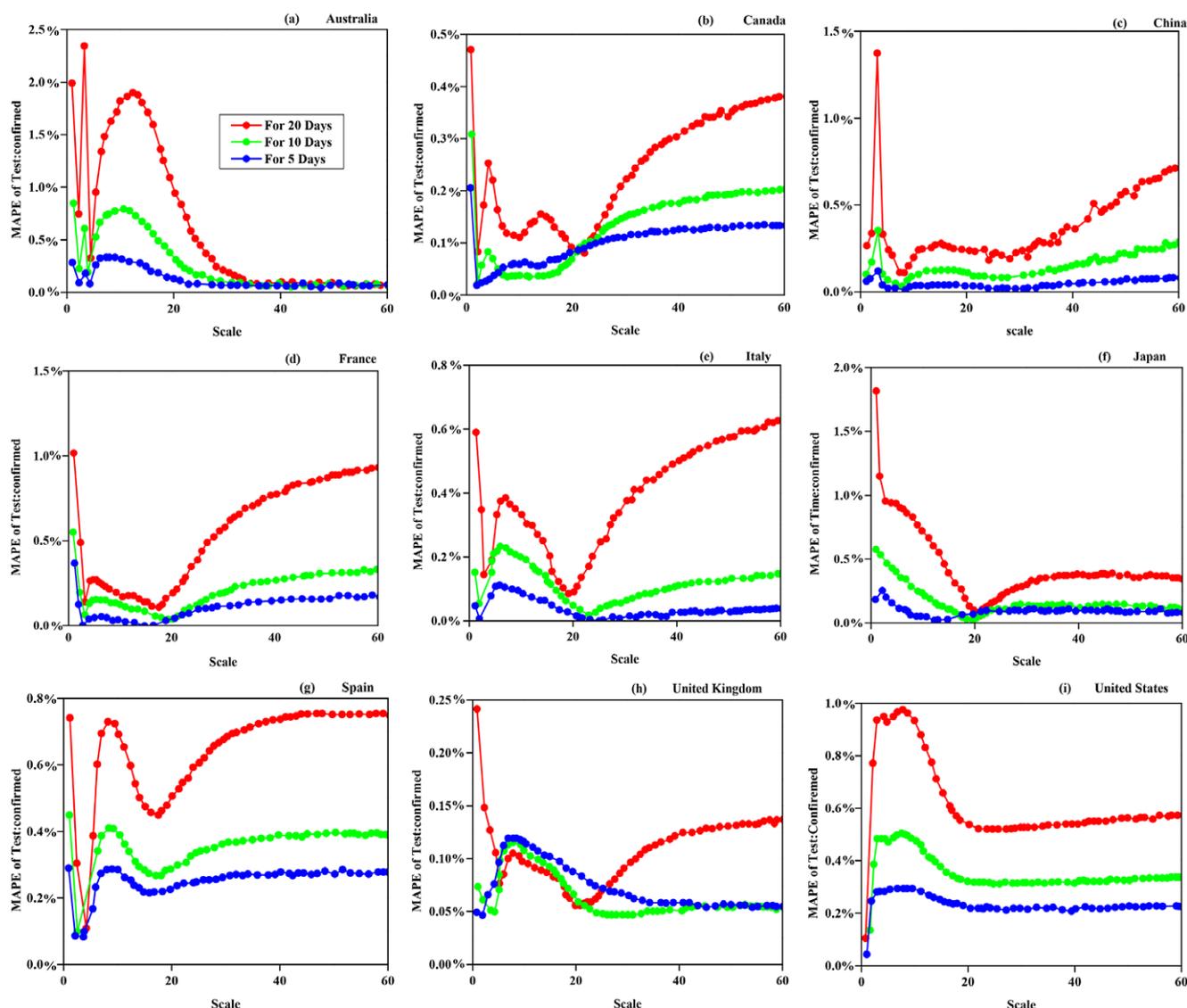$$\eta(d) = \eta_0 \times \left(1 - \frac{r}{R}\right) = 0.1 \left(1 - \frac{r}{100}\right) \qquad (24)$$

Figure 8. The variation of predictive performance MAPE with increasing *scale* for confirmed cases forecasting for nine countries.

From the above equation, denotes the initial learning value, the iterative number is represented as *r*.

For the data self-driven ESN-DSS, a modified framework is utilized [25] to update reservoir neuronal states.

$$r(t+1) = (1-\alpha)r(t) + \alpha \tanh(\mathbf{W_{in}}u(t+1) + \mathbf{W}r(t) + v(t) + d),$$
$$(25)$$

where activation functions of reservoir states uniformly adopt hyperbolic tangent functions (tanh) for N neurons. In the training phase, the input of the network $u(t+1)$ is the actual training sample output at time step t, not the predicted output, namely $u(t+1) = y(t)$. In the test process, the input $u(t+1)$ is the predicted output of self-driven ESN with respect to the time step t, namely $u(t+1) = y(t)$. The weight matrix of input connections W_{in} is randomly assigned values based on a uniform distribution, with a value range of $[-1,1]$. The weight matrix of neuronal internal connections W is a random sparse

matrix with sparse connectivity of 10% from a uniform distribution. The value range for W is $[-1,1]$, and the spectral radius of W generally satisfies $\rho(W) \leq 1$. $v(t)$ is an $N \times 1$ random noise vector from a uniform distribution over $[-1 \times 10^{-7}, 1 \times 10^{-7}]$. d is an $N \times 1$ bias vector,

$$d = \operatorname{arctanh}(r^*) - \mathbf{W}r^*,$$
$$(26)$$

where r^* is an $N \times 1$ random vector from a uniform distribution over $(-1, -0.8]$ and $[0.8, 1)$. The predicting output is modified as

$$\widehat{y}(t+1) = \mathbf{W_{out}}(u(t+1), r(t+1)).$$
$$(27)$$

It is worth noting that $u(t+1) = y(t)$ during the training period, while $u(t+1) = y(t)$ in the test phase. The reservoir state matrix R in the training process is

$$\mathbf{R} = \begin{bmatrix} y(T_{initial}) & y(T_{initial}+1) & \cdots & y(T_{train}-1) \\ r(T_{initial}+1) & r(T_{initial}+2) & \cdots & r(T_{train}) \end{bmatrix}.$$
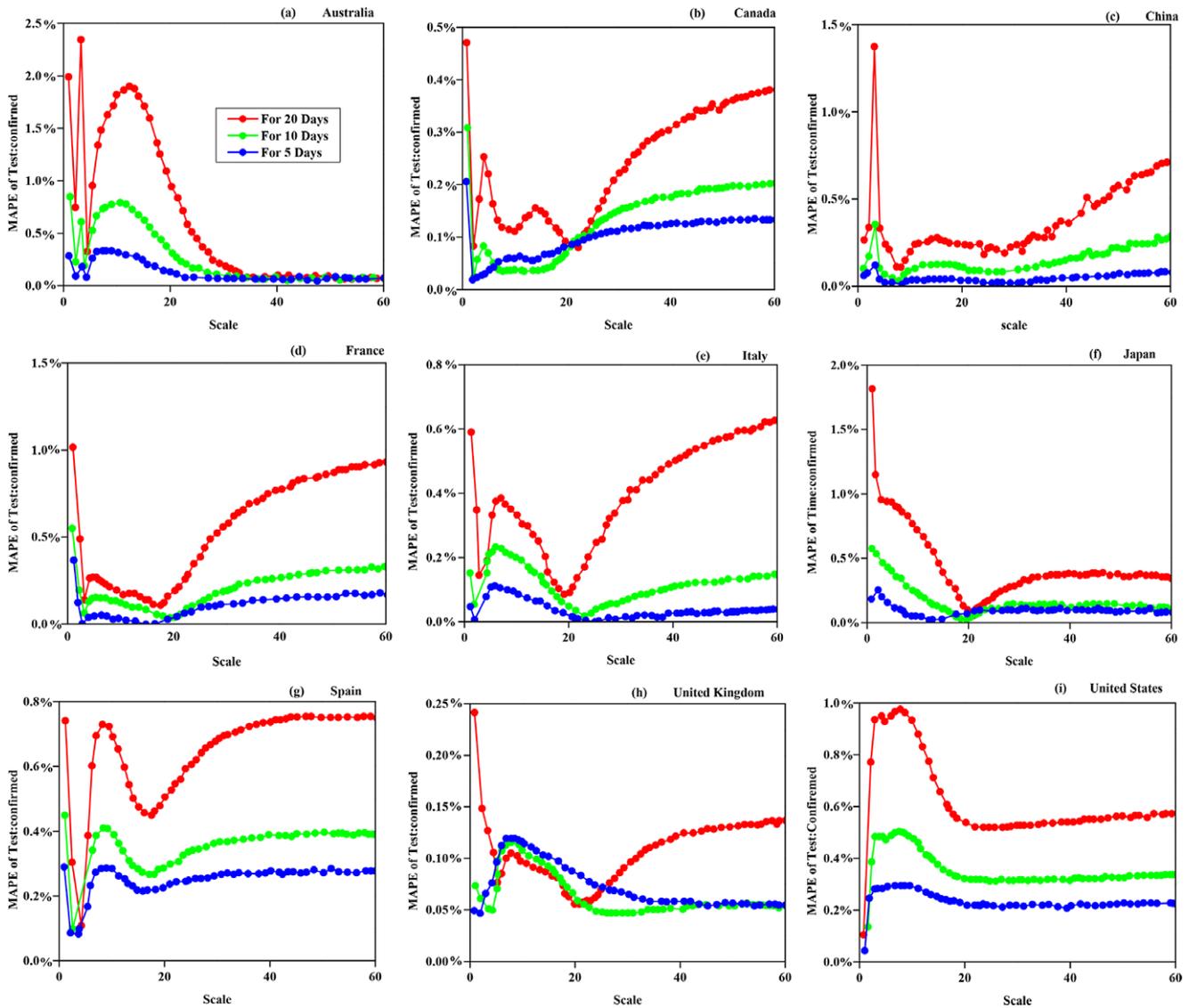$$(28)$$

Figure 9. The change of MAPE with increasing the number of reservoir neurons *N* for confirmed cases forecasting for nine countries.

After the output weight matrix W_{out} is obtained according to Equation (6), the self-driven network is ready for use.

4. Experimental results

This section shows the performance evaluation of forecasting epidemics using the proposed model. Further, the method is analysed by several metrics which include the mean absolute error (MAE), RMSE, explained variance (EV), RMSLE, and mean absolute percentage error (MAPE). Then, the output of the proposed model is compared with all other methods. The outputs are analysed in the following sections.

4.1. Evaluation indices

In this article, to evaluate the effectiveness and accuracy of the forecast model based on the data self-driven ESN, five performance indicators are analysed. The MAE, RMSE, EV, RMSLE, and MAPE are expressed mathematically as follows, respectively.

RMSE: This is used to evaluate the prediction performance based on the Euclidean distance that extracted the true values. The average variations are estimated based on predicted and actual values.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y(i)-\widehat{y}(i)\right)^2}, \tag{29}$$

MAE: It determined the total error arised among different directions, and the predicted values showed the alternate measurement in prediction process.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y(i)-\widehat{y}(i)\right|, \tag{30}$$

MAPE: It is used to validate the average range of the absolute function among varied prediction process that extracted the error and obtained an absolute prediction.

$$MAPE = \left(\frac{100}{n} \times \sum_{i=1}^{n} \left| \frac{y(i) - \widehat{y}(i)}{y(i)} \right| \right) \%, \qquad (31)$$

EV: It measures the total variance range of each independent variable, and the ratio is validated from the overall gathered data variables.

$$EV = 1 - \frac{\text{var}\left(\widehat{Y} - Y\right)}{\text{var}(Y)}, \qquad (32)$$

RMSLE: It compared the predicted and actual values for evaluating the less penalized function based on real range.

$$RMSLE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\ln(y(i)) - \ln(\widehat{y}(i))\right)^2}. \qquad (33)$$

From the above equations, n is the number of samples (days), $y(i)$ is the ith actual values, and $y(i)$ is the corresponding predicted values, and var is the variance of the time series computed by the function $var()$ of MATLAB. RMSE and MAE are absolute error, while MAPE and RMSLE are relative error. RMSLE and RMSE are similar, except that RMSLE is computed at the logarithmic scale, and thus the data size is ignored. For the best model, the expected values of RMSE, MAE, MAPE, and RMSLE are zero, and an EV closer to one indicates more accurate prediction performance.

4.2. Parameter settings

In this study, the sample data that come from the CSSE at Johns Hopkins University is the number of daily cumulative confirmed cases of 881 days from 22 January 2020 to 20 June 2022, where the samples of 861 days from 22 January 2020 to 31 May 2022, are used to train the model to compute the output weight matrix W_{out}, and then forecasting the total number of cumulative cases that are confirmed for the next 20 days from 1 to 20 June 2022, that is, the number of training and test samples are $T_{train} = 861$ and $T_{test} = 20$ respectively.

COVID-19 time series of daily cumulative confirmed cases is one-dimensional univariate. Therefore, for our model, the number of the input and output units is $K = L = 1$. The number of reservoir neurons is set as $N = 800$. The reservoir starts with all zero network states, namely $R(0) = (0, 0, \ldots, 0)$. According to the state forgetting property of the reservoir, the effects of the initial reservoir state have died out after $T_{initial}$ steps so that an initial transient of 100 steps, namely $T_{initial} = 100$, is discarded. Table 1 shows the assigned parameters.

4.3. Forecast results

In this study, the temporal evolution of daily cumulative confirmed cases from nine considered countries is predicted, including the United Kingdom, Canada, Italy, France, Australia, China, the United States, Spain, and Japan. In the following simulations, without special mention, all data of testing results are restored original scale according to the inverse function.

The training and test results of prediction confirmed cases in the nine countries are given in Figure 5. To observe the test results more clearly, the sub-graphs of Figure 5 illustrate the magnifying test results of forecasted confirmed cases based on the testing data of 20 days. Figure 6 represents the evolutions of the absolute percentage error (APE) of prediction confirmed cases with time during the

testing period for nine countries. The errors APEs gradually increase with the increase of forecast days in most countries can be observed, while there are exceptions, such as Japan and the United Kingdom whose errors APEs fluctuate, as shown in Figure 6(f, h). This is because our research belongs to short-term forecasts. If the number of prediction days increases, the forecast errors will also increase. Overall, the prediction errors in APEs do not exceed 2% for all considered countries. The results exhibit good forecasting performance during the testing period.

Figure 7 illustrates the error histograms of forecasted confirmed cases for nine countries based on the testing data of 5, 10, and 20 days. The blue bar is the mean error of forecasted 20 days, namely predicting the mean error from 1 to 20 June 2022. The orange-red bar is the mean error of forecasted 10 days from 1 to 10 June 2022. The mean error of forecasted 5 days is expressed by the yellow bar, namely, testing means error from 1 to 5 June 2022. The results indicate that the United States has the largest absolute error on RMSE and MAE, as shown in Figure 7(a, b) because its basic real data are the largest, as shown in Figure 1. Australia has a relatively large relative error on MAPE and RMLSE, as shown in Figure 7(c, d). Although Australian relative error is the largest, the largest MAPE does not exceed 1%, as shown in Figure 7(c). The detailed evaluation indices of error for confirmed cases forecasting during the testing period are shown in Table 2.

So far, many methods of deep learning are used in forecasting the time series evolution of the COVID-19 epidemic. For the forecast results and errors, the comparison of the self-driven ESN model is carried out with the other deep learning prediction models which include standard RNN, bidirectional LSTM (BiLSTM), GRU, LSTM, and VAE [15]. Table 3 depicts the comparison results.

In the comparative simulation experiment, the model is trained and tested to forecast the time evolution of the cases which are confirmed at the time scale that is the same as the contrastive five methods. Training samples are the time series of 131 days from 22 January 2020 to 31 May 2020. Then, the number of cumulative confirmed cases is forecasted for the next 17 days from 1 to 17 June 2020, for six countries, including Australia, China, France, Italy, Spain, and the United States. Due to few training samples, in our model, the initial transient is set as $T = 40$, namely the first 40 steps of the reservoir state are discarded. Other parameters remain unchanged. In Table 3, the best prediction results of the different evaluation indices are marked in bold among the six methods compared. It is obvious from the comparison results that our model has better predictive performance compared with other models in aspect of some performances, including MAE, MAPE, and RMSE, whereas the prediction accuracy of our model is even five orders of magnitude better than some models. However, in the aspect of EV and RMSLE, the performances of our model are sometimes slightly inferior to those of VAE.

4.4. Influences of model parameters

In this section, the impact of the parameters of the model on prediction performance is analysed. These parameters include the data scaling ratio, number of reservoir neurons, leakage rate, and spectral radius of the connection matrix, *W*.

To maintain the echo states of the network, the COVID-19 time series need to be squashed into a range $[-1, 1]$. By training the model using training data, the time series evolution of the next 20 days is forecasted. In principle, the scaling factor *scale* cannot be less than the growth rate of the predictive 20-day interval. However,
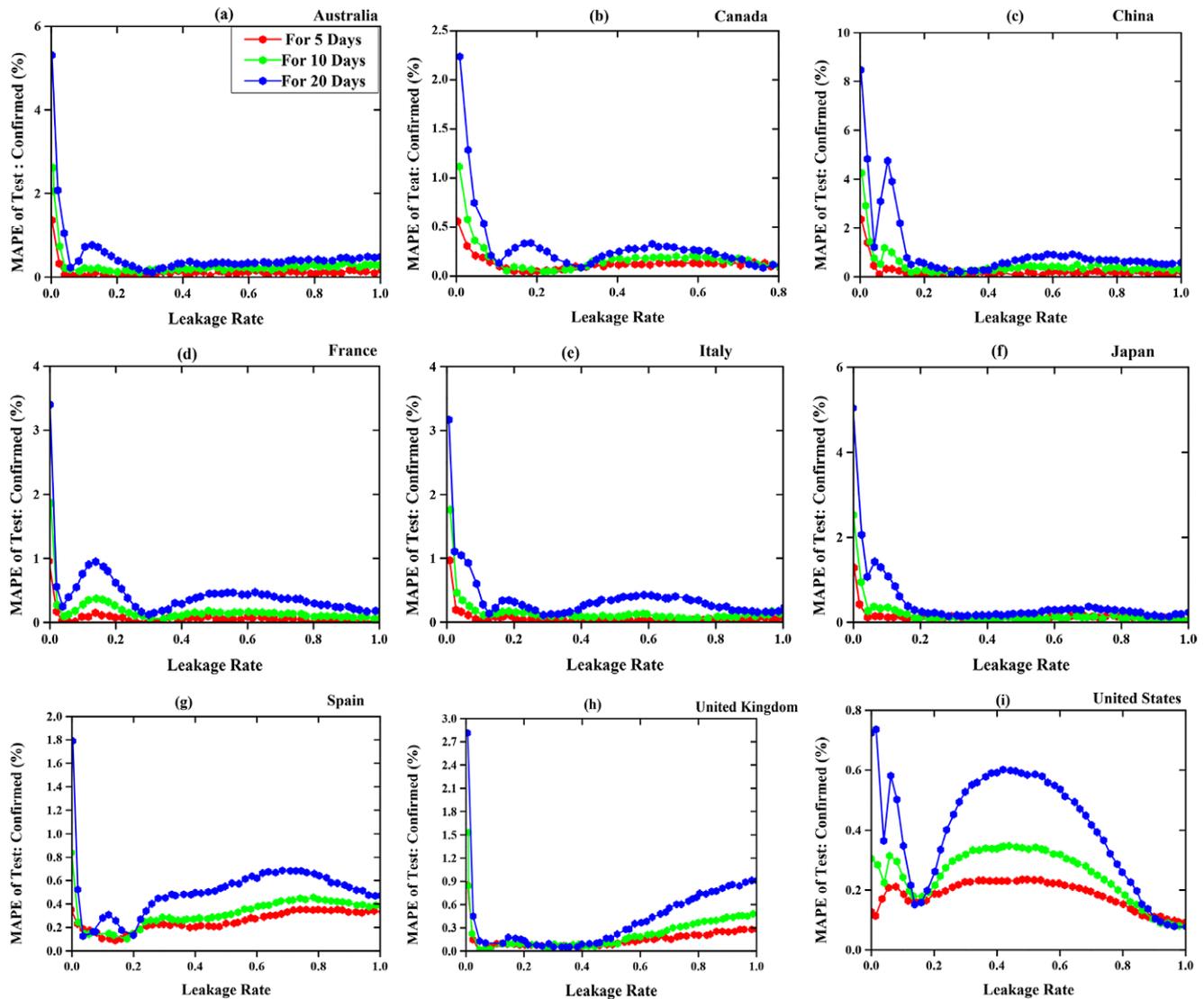
Figure 10. The change of MAPE with increasing leakage rate α for confirmed cases forecasting for nine countries.

the test data for the next 20 days are assumed to be unknown. Therefore, it is not easy to evaluate the value of *scale*.

The value range *scale* is kept from 1 to 60, and the other parameters remain unchanged. MAPE of forecast results of the number of confirmed cases based on the testing data of 5 days (pink line), 10 days (green line), and 20 days (blue line) are shown in Figure 8. We can observe that the prediction performance is very unstable when the value *scale* is relatively small. This is because the training data are squashed into a range of $[-1, 1]$, while relatively small *scale* may induce that the inputs and outputs of self-driven ESN are greater than one during the test phase. This may affect the echo state of the network to lead to unstable prediction performance. When the value *scale* is slightly larger, the optimal or suboptimal *scale* exists to optimize predictive performance. For the different countries, the optimal *scale* is different, as shown in Figure 12(a).

The impact of network size N is analysed, namely the number of reservoir neurons, on the testing error. The forecast results are displayed in Figure 9. We can see that a bigger network size does not always bring better results. Smaller or bigger network sizes may lead to under-fitting or over-fitting. For some cases, there is an optimal network size to improve prediction performance. However, for the different countries, the optimal network size is different. For example, the predicted performance does not change much when the network size continues to increase from 500 for China, as shown in Figure 9(c), and similar situations are also shown in Figure 9(g, i). The optimal network size for the different countries is collected in Figure 12(b).

The leakage rate α is to control the update speed of the reservoir neuron states and satisfies $0 \leq \alpha \leq 1$. The larger the leakage rate is, the faster the update speed is. The smaller leakage rate means that the reservoir states at the previous time step have a greater influence on reservoir status updates at this moment. The influence of the leakage rate on forecast accuracy is studied, as shown in Figure 10. The optimal leakage rate is determined, which promotes predictive performance almost distributes between 0.2 and 0.4 for confirmed cases forecasting in nine countries, as shown in Figure 12(c). For the United States, in Figure 10(i), there is an optimal and a suboptimal value of the leakage rate, and the suboptimal value is close to 0.2.

The influence of the spectral radius of the reservoir neurons connection matrix W on forecast accuracy is analysed. In principle,
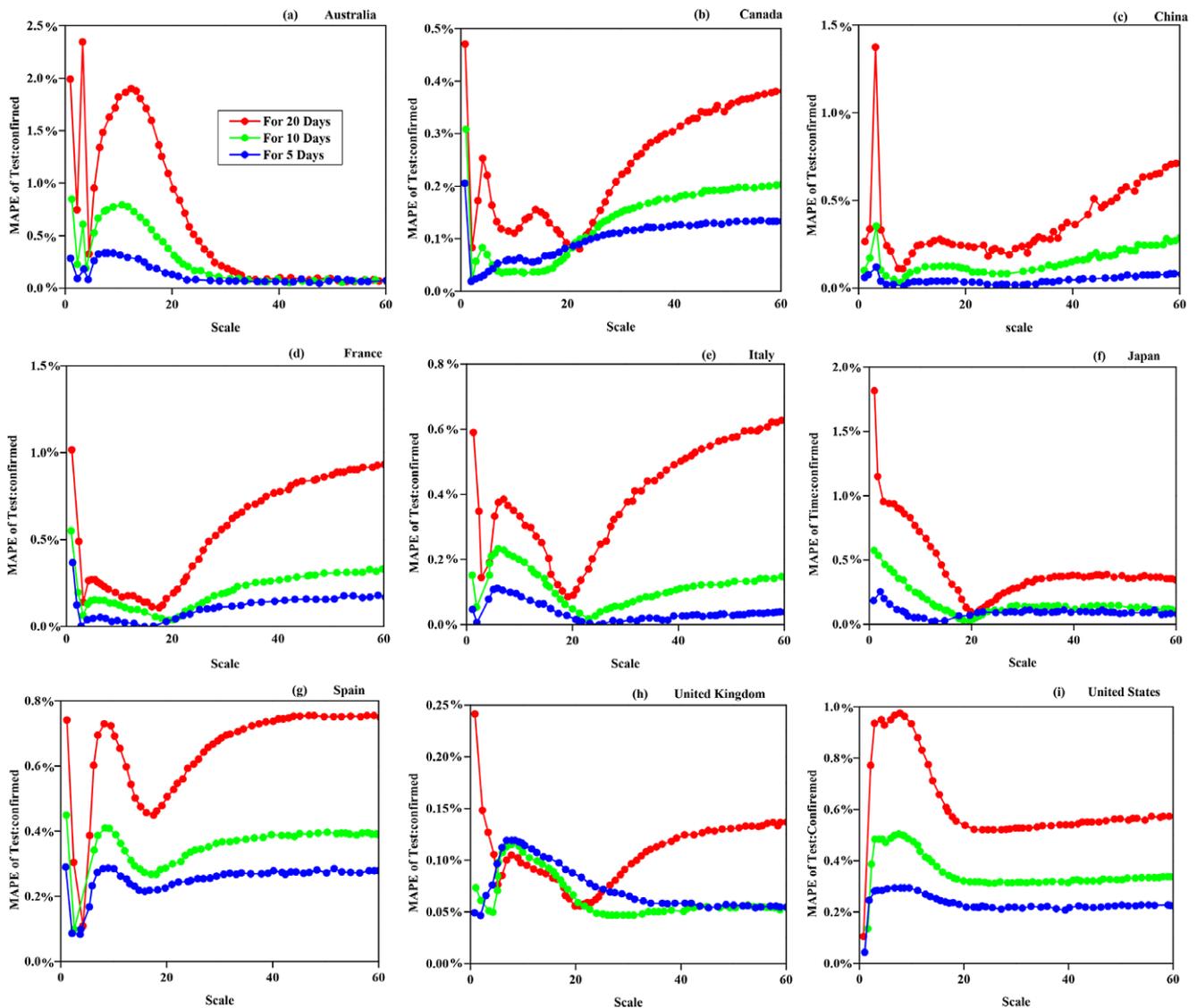
Figure 11. The variation of MAPE with increasing spectral radius $\rho(W)$ for confirmed cases forecasting for nine countries.

the spectral radius satisfies $\rho(W) < 1$ to hold the echo state of networks, but this condition is excessively restrictive. If a network has a weight matrix $W\rho(W) \geq 1$, the echo state property can hold nonetheless if the input not containing the null input [18]. Therefore, in the simulation experiment, the spectral radius is set over $(0,2)$. Figure 11 illustrates the change of the test error MAPE with the enlargement of the spectral radius. In Figure 11, for most countries, the optimal spectral radius is closer to one, while prediction performances decrease with increasing the spectral radius when the spectral radius is greater than one. However, there are exceptions, as shown in Figure 11(g, i), and the larger the spectral radius, the better the forecast accuracy. The values of the optimal spectral radius to optimize prediction performance are shown in Figure 12(d).

The spectral radius reflects the short-term memory capacity of the network. The bigger spectral radius means that the short-term memory capacity of the self-driven ESN is longer, and vice versa. Figure 11(g, i) illustrates that the LSTM (or more historical information) can help the self-driven ESN to better simulate and predict the future temporal information for Spain and the United State.

In a word, by analysing the impact of the model parameters (including the data scaling ratio, the number of reservoir neurons spectral radius of the connection matrix of reservoir neurons) on prediction performance and leakage rate, these parameters can be set adaptively as an appropriate value or in an appropriate range to reduce the prediction error of the self-driven ESN model.

5. Conclusion

The COVID-19 pandemic has negatively affected public health and economic development all over the world and brought heavy burdens to healthcare systems in many countries. In our study, the data self-driven ESN is established to predict the COVID-19 time series. The data self-driven ESN was created to form the closed feedback loop by replacing the input with the output. The nine different countries are selected to forecast the time evolution of daily cumulative confirmed cases. By comparison with other deep learning forecasting models at the same time scale, our model showed excellent prediction performances. Moreover, the influence of the model parameters on forecast accuracy is analysed, and
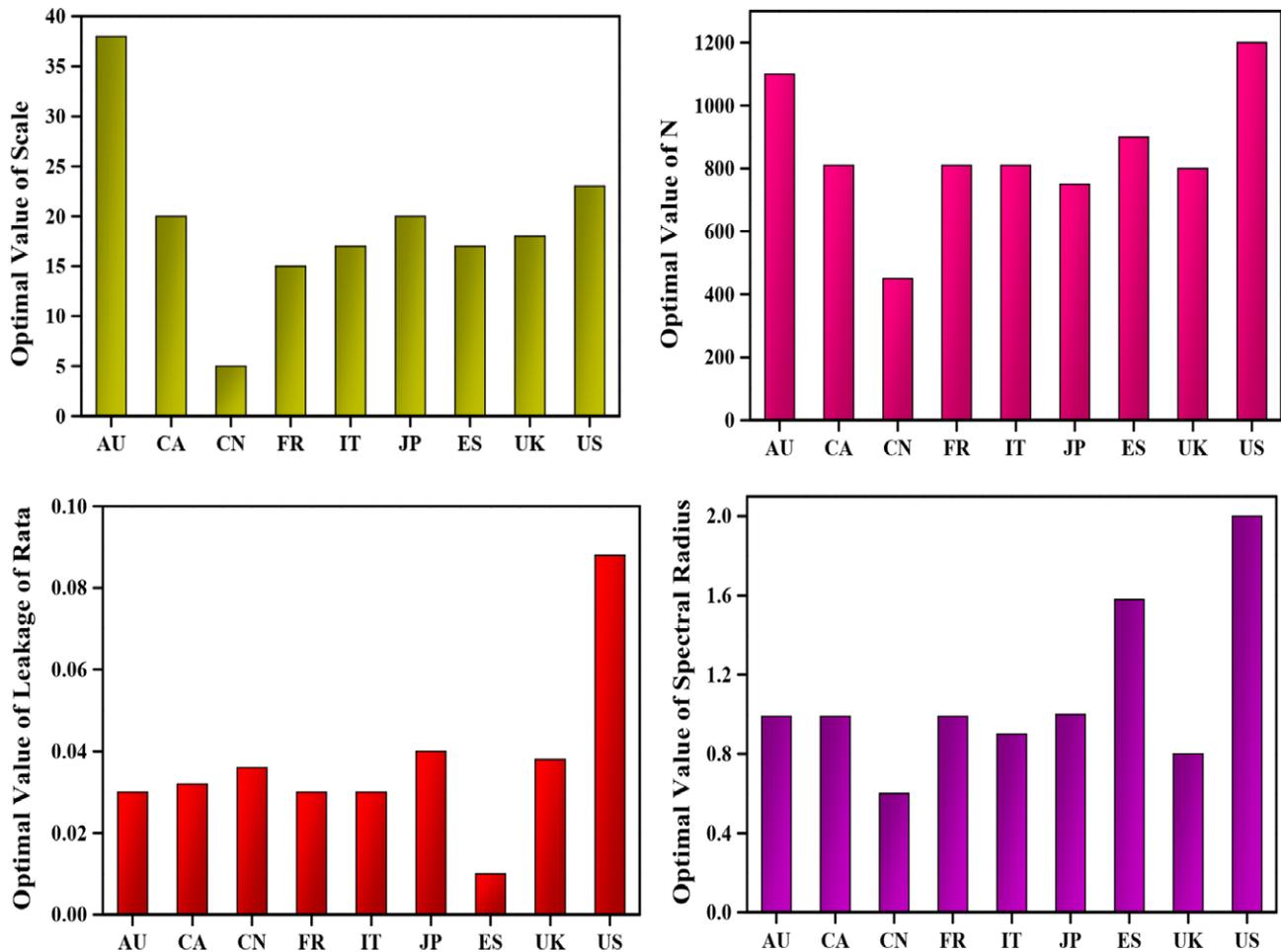
Figure 12. The optimal parameter to minimize the test error MAPE of confirmed cases forecasting for the next 20 days in nine countries.

found that the forecasting error can be reduced by selecting the appropriate data scaling ratio, the number of reservoir neurons, the leakage rate, and the spectral radius of the connection matrix of reservoir neurons.

To summarize, through the present study, appropriate parameters can be adaptively selected to improve the forecast accuracy of the self-driven ESN model. The results could help governments or medical institutions to monitor the current situation and to take opportune precautionary strategies to prevent further spread. Additionally, this study is not only limited to the prediction of the COVID-19 epidemic but also applicable to other nonlinear time series prediction problems. It failed to extract the actions taken by the government that prevent to analyse the total death and cured cases. It did not validate the result with non-parametric models that extract the uncertainty issue obtained in the prediction process. Even though, various achievements have been made, several enhancements will be made in the future for the betterment of this work, which includes the development of a novel technique to detect COVID-19 cases in a short time, thereby accessing the performance enhancement. Furthermore, it extended to consider the list of predicted and mortality cases from an individual country regarding COVID-19 pandemic.

Author contribution. WW, QL and JW authors agreed on the content of the study. WW, QL and JW collected all the data for analysis. JW agreed on the methodology. WW, QL and JW completed the analysis based on agreed steps. Results and conclusions are discussed and written together. All authors read and approved the final manuscript.

Competing interest. The authors declare no conflict of interest.

Ethical standard. This article does not contain any studies with human or animal subjects performed by any of the authors. Informed consent was obtained from all individual participants included in the study.

References

[1] **Viana R**, **Moyo S**, **Amoako DG** et al. (2022). Rapid epidemic expansion of the SARS-CoV-2 omicron variant in southern Africa. *Nature* **603**(7902), 679–686. https://doi.org/10.1038/s41586-022-04411-y

[2] **Adrielle Dos Santos L**, **Filho PGG**, **Silva AMF** et al. (2021). Recurrent COVID-19 including evidence of reinfection and enhanced severity in thirty Brazilian healthcare workers. *Journal of Infection* **82**(3), 399–406. https://doi.org/10.1016/j.jinf.2021.01.020

[3] **Leung K**, **Wu JT**, **Leung GM** (2021). Real-time tracking and prediction of COVID-19 infection using digital proxies of population mobility and mixing. *Nature Communications* **12**(1), 1501. https://doi.org/10.1038/s41467-021-21776-2

[4] **Hussain H**, **Ozair M**, **Ali F** et al. (2021). Sensitivity analysis and optimal control of COVID-19 dynamics based on SEIQR model. *Results in Physics* **22**, 103956. https://doi.org/10.1016/j.rinp.2021.103956

[5] **Firmino PRA**, **de Sales JP**, **Goncalves Junior J** et al. (2020). A non-central beta model to forecast and evaluate pandemics time series. *Chaos Solitons & Fractals* **140**, 110211. https://doi.org/10.1016/j.chaos.2020.110211

[6] **Arias Velasquez RM**, **Mejia Lara JV** (2020). Forecast and evaluation of COVID-19 spreading in USA with reduced-space Gaussian process regression. *Chaos Solitons & Fractals* **136**, 109924. https://doi.org/10.1016/j.chaos.2020.109924

[7] **Sahin U**, **Sahin T** (2020). Forecasting the cumulative number of confirmed cases of COVID-19 in Italy, UK and USA using fractional nonlinear grey Bernoulli model. *Chaos Solitons & Fractals* **138**, 109948. https://doi.org/10.1016/j.chaos.2020.109948

[8] **Feroze N** (2020). Forecasting the patterns of COVID-19 and causal impacts of lockdown in top five affected countries using Bayesian structural time series models. *Chaos Solitons & Fractals* **140**, 110196. https://doi.org/10.1016/j.chaos.2020.110196

[9] **Santosh KC** (2020). AI-driven tools for coronavirus outbreak: Need of active learning and cross-population train/test models on Multitudinal/multimodal data. *Journal of Medical Systems* **44**(5), 93. https://doi.org/10.1007/s10916-020-01562-1

[10] **Gomez-Cravioto DA**, **Diaz-Ramos RE**, **Cantu-Ortiz FJ** et al. (2021). Data analysis and forecasting of the COVID-19 spread: a comparison of recurrent neural networks and time series models. *Cognitive Computation* **16**, 1794–1805. https://doi.org/10.1007/s12559-021-09885-y

[11] **Hu HL**, **Wang L**, **Lv SX** (2020). Forecasting energy consumption and wind power generation using deep echo state network. *Renewable Energy* **154**, 598–613. https://doi.org/10.1016/j.renene.2020.03.042

[12] **Ghosh S**, **Senapati A**, **Mishra A** et al. (2021). Reservoir computing on epidemic spreading: A case study on COVID-19 cases. *Physical Review E* **104**(1–1), 014308. https://doi.org/10.1103/PhysRevE.104.014308

[13] **MacIntyre CR**, **Chen X**, **Kunasekaran M**, **Quigley A**, **Lim S**, **Stone H**, **Paik HY**, **Yao L**, **Heslop D**, **Wei W**, **Sarmiento I** (2023). Artificial intelligence in public health: The potential of epidemic early warning systems. *Journal of International Medical Research* **51**(3), 03000605231159335.

[14] **Uzun Ozsahin D**, **Precious Onakpojeruo E**, **Bartholomew Duwa B**, **Usman AG**, **Isah Abba S**, **Uzun B** (2023). COVID-19 prediction using black-box based Pearson correlation approach. *Diagnostics* **13**(7), 1264.

[15] **Arora R**, **Agrawal A**, **Arora R**, **Poonia RC**, **Madaan V** (2021). Prediction and forecasting of COVID-19 outbreak using regression and ARIMA models. *Journal of Interdisciplinary Mathematics* **24**(1), 227–243.

[16] **Hasan A**, **Putri ER**, **Susanto H**, **Nuraini N** (2022). Data-driven modeling and forecasting of COVID-19 outbreak for public policy making. *ISA Transactions* **124**, 135–143.

[17] **Saqib M** (2021). Forecasting COVID-19 outbreak progression using hybrid polynomial-Bayesian ridge regression model. *Applied Intelligence* **51**(5), 2703–2713.

[18] **Dash S**, **Chakravarty S**, **Mohanty SN**, **Pattanaik CR**, **Jain S** (2021). A deep learning method to forecast COVID-19 outbreak. *New Generation Computing*, **39**(3–4), 515–539.

[19] **Namasudra S**, **Dhamodharavadhani S**, **Rathipriya R** (2021). Nonlinear neural network based forecasting model for predicting COVID-19 cases. *Neural Processing Letters* **55**, 1–21.

[20] **Ketu S**, **Mishra PK** (2021). Enhanced Gaussian process regression-based forecasting model for COVID-19 outbreak and significance of IoT for its detection. *Applied Intelligence* **51**, 1492–1512.

[21] **John Hopkins University** (2022) COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University. https://github.com/CSSEGISandData/COVID-19 (accessed 20 June 2022).

[22] **Jaeger H** (2001). The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. Bonn, Germany: German National Research Center for Information Technology GMD Technical Report **148**(34), 13.

[23] **Jaeger H**, **Lukosevicius M**, **Popovici D** et al. (2007). Optimization and applications of echo state networks with leaky-integrator neurons. *Neural Networks* **20**(3), 335–352. https://doi.org/10.1016/j.neunet.2007.04.016

[24] **Su MC**, **Chen JH**, **Utami AM** et al. (2022). Dove swarm optimization algorithm. *IEEE Access* **10**, 46690–46696.

[25] **Kim JZ**, **Lu ZX**, **Nozari E** et al. (2021). Teaching recurrent neural networks to infer global temporal structure from local examples. *Nature Machine Intelligence* **3**(4), 316–323.