



Time-series benchmarks based on frequency features for fair comparative evaluation

Zhou Wu¹ · Ruiqi Jiang¹

Received: 18 August 2022 / Accepted: 28 March 2023 / Published online: 22 April 2023
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2023

Abstract

Time-series prediction and imputation receive lots of attention in academic and industrial areas. Machine learning methods have been developed for specific time-series scenarios; however, it is difficult to evaluate the effectiveness of a certain method on other new cases. In the perspective of frequency features, a comprehensive benchmark for time-series prediction is designed for fair evaluation. A prediction problem generation process, composed of the finite impulse response filter-based approach and problem setting module, is adopted to generate the NCAA2022 dataset, which includes 16 prediction problems. To reduce the computational burden, the filter parameters matrix is divided into sub-matrices. The discrete Fourier transform is introduced to analyze the frequency distribution of transformed results. In addition, a baseline experiment further reflects the benchmarking capability of NCAA2022 dataset.

Keywords Evaluation dataset · Frequency domain · NCAA2022 · Time-series prediction

1 Introduction

Time-series prediction has played important roles in the analysis of future markets, meteorology, pandemic, and other industrial/customer applications, in which historical statistics would be regarded as time-series signals to build computational prediction models. Related applications could be categorized into single-step [1], multi-step [2], multi-timescale [3], multivariate [4] prediction, and missing data imputation [5]. Many machine learning techniques have been proposed, including statistical methods [6], artificial neural network [7], support vector machine (SVM) [8], and other approaches.

To further improve the prediction performance of specific tasks, researchers design advanced methods according to features of time-series data. For example, the characteristics of financial data include volatility, irregularity, noise, and inertia, so modified fuzzy systems [9],

hybrid neural networks [10, 11], and advanced filters [12, 13] are observed with effectiveness. The wind power data have the inherent nature of intermittence and randomness, so researchers have studied data pre-process techniques [14] and mined spatial correlation to improve prediction accuracy [15].

It could be observed that existing prediction approaches are proposed based on specific application scenarios. According to the “no free lunch” theory [16], a certain approach has satisfactory performance on some time-series scenarios and must be ineffective on other ones. How do we fairly evaluate these approaches for general prediction tasks? A comprehensive set of time-series benchmarks is necessary for covering most data characteristics, and an online competition could be held by inviting volunteer researchers to solve the same benchmarks. The performance of each competing approach could be analyzed, and overall rankings could indicate how to choose an appropriate approach for fitting a certain kind of characteristics. In the M3 competition [17], the organizers summarize several previous competitions, and propose a business-related dataset of 3033 benchmarks, which include micro, industrial, and macrosenarios with different time scale, e.g., yearly, quarterly, and so on. The results of M3 competition illustrate that the time-series prediction algorithms

✉ Zhou Wu
zhouwu@cqu.edu.cn

Ruiqi Jiang
jiang_ruiqi@cqu.edu.cn

¹ School of Automation, Chongqing University, Shazheng Street, Chongqing 400044, China

perform variously in different datasets [18]. The M3 competition organizers expect this large-volume dataset analyzable, but neglect the fairness of feature evaluation. In the M4 competition, the organizers continued to increase the volume of dataset [19]. As extensions of M3 and M4, NN3 and NN5 competitions focus on evaluating prediction approaches through single or double timescale datasets [20, 21].

In addition, there are certain proposed datasets to promote the developments of related research techniques [22]. Pandemic prediction, about saving millions of lives, is one of the topic research areas, and various datasets are presented to facilitate researchers [23–26]. Besides, as energy topics get continuous attention from society, researches on energy demand and consumption are rapidly increasing. Hence, many energy-related datasets are proposed to facilitate researchers to study the improvement of energy efficiency [27–30]. Apart from the aforementioned areas, many data-driven methods also have related time-series datasets for researchers, e.g., solar flare research [31], plants detection [32], remote sensing [33], and waste stream [34].

However, existing datasets and competitions have only considered large-volume datasets to evaluate the performance of prediction approaches, and few studies have categorized the features of time-series data, which causes the failure of characteristic analysis. In fact, prediction performance has significant variance with respect to data characteristics, especially for deep learning approaches [35]. Neural network-based methods, widely adopted in time-series prediction, are sensitive to data features in low frequency [36]. To further improve the prediction performance of neural networks, the learning ability of high-frequency features should be enhanced [37]. However, for the datasets proposed in time-series competitions, frequency features are seldom considered and categorized. Moreover, for complex time-series prediction, although certain studies decompose the data in frequency domain [38–41], they ignore the impact of different frequency features to their approaches. Therefore, a new method based on the finite impulse response (FIR) filter is developed to generate a set of 16 benchmarks, which is used in the time-series competition held at the 2022 international conference of neural computing with advanced applications (NCAA2022).

The NCAA2022 dataset, including 16 time-series prediction benchmarks, is proposed to emphasize the impacts of frequency characteristics on different prediction models. In the NCAA2022 competition dataset, 4 original time-series instances are collected, and each problem is transformed into low-pass, high-pass, band-pass, and band-stop benchmarks. A new generation method is proposed based on the adaptive design of FIR filter, and the transform

process is simplified by sub-matrices to reserve computation cost.

The rest of this paper is organized as follows: The original time-series instances are introduced in Sect. 2. In Sect. 3, the generation methodology of new benchmarks is proposed. In Sect. 4, the evaluation metric of NCAA2022 dataset is described, and the baseline performance and comparative results are presented. Finally, the paper is concluded in Sect. 5.

2 Original instances

The original instances are required to reflect comprehensive characteristics, so coronavirus (COVID-19) pandemic data, Dow Jones Industrial average indices, and wind speed data, are chosen as original instances, which are available in [42–44]. Furthermore, the Lorenz system, a classical benchmark for time-series prediction [45, 46], is also chosen for integrating the chaotic characteristics.

The market prediction is a worldwide hot topic, and many machine-learning approaches are applied to predict the indices. Generally, there is a nonlinear relation between the future and past indices, so the fitting approaches for the nonlinear relation are widely applied [47, 48].

The closing indices of Dow Jones Industrial Average from December 25, 1979, to November 17, 2021, are chosen as the stock raw data [43]. The stock raw data are one-dimension with a 10567 length. Researchers in market prediction are mainly considering historical indices [49, 50], other related factors are neglected in the stock raw instance.

With the increasing concern about greenhouse effects, renewable energy prediction has become the key point to improving the efficiency of power generation. Due to the volatile feature of wind speed, the prediction of wind power is complicated and challenging. Existing approaches could be categorized into physical and statistical models [51]. To handle volatility, neural network and data processing methods are often combined in most studies [52].

The wind speed raw data are congregated by wind speed data collected at two sites, i.e., Ames and Dubuque, from January 1, 2010, to December 28, 2011. The wind data have 2 dimensions, and each dimension has the length of 15000 [44]. Both two sites are located in Iowa, America, and there is a spatial correlation of wind speed between the two sites. Certain advanced machine learning models have taken advantage of the spatial correlation to improve the prediction performance [53], so data from two sites are provided as references in the NCAA2022 competition.

In order to reduce the influence of COVID-19 outbreak, researchers are focusing on prediction and control methods for pandemic spreading. Due to the short duration of

COVID-19, there is a limited amount of pandemic-related data. As a result, the prediction approaches need to have the capability of few-shot or transfer learning [54, 55].

The number of daily confirmed cases in the USA, UK, Canada, and Germany from January 22, 2020, to November 20, 2021, are chosen as pandemic raw data [42]. Thus, the COVID-19 data have 4 dimensions, and each dimension has the length of 669. As the confirmed data include certain correlations on geographical information [56], COVID-19 data over 4 countries are given for references in the NCAA2022 competition.

Chaotic systems commonly exist in the real world, so the chaos is considered in time-series prediction approaches [57, 58]. The Lorenz system is chosen as the chaotic raw data, which has nonlinearity and initial condition sensitivity. The canonical form of Lorenz system is expressed as

$$\begin{cases} \dot{z}_1(t) = a(z_2(t) - z_1(t)) \\ \dot{z}_2(t) = z_1(t)(b - z_3(t)) - z_2(t) \\ \dot{z}_3(t) = z_1(t)x_2(t) - cz_3(t), \end{cases} \quad (1)$$

where a, b, c are chaotic parameters and $z_1(t), z_2(t), z_3(t)$ are three dimensions of Lorenz system at time t . In this paper, the parameters and initial conditions are set as $a = 10, b = 28, c = 3, z_1(0) = 0, z_2(0) = 1, z_3(0) = 0$. The chaotic raw data have 3 dimensions. Assume the sampling period and duration are 0.01 s and 150 s, respectively, and each dimension has the length of 15000. The chaotic data are simulated based on the Runge–Kutta solutions of Lorenz system [59].

The information of four chosen raw data is shown in Table 1.

3 Methodology of dataset generation

To generate a comprehensive set of benchmarks with respect to different frequency features, an FIR filter-based approach is developed. As illustrated in Fig. 1, the generation process includes an FIR filter and problem setting module. The FIR filter is responsible to transform each raw data into four ones with low-pass, high-pass, band-pass,

and band-stop features. In the problem setting module, the transformed data are processed into prediction problems.

3.1 Data transformation

A transformation is needed to generate required data from stock, COVID-19, wind speed, and chaos data. As shown in Fig. 2, the transform process could be denoted as

$$y(t) = f(x), \quad (2)$$

where x is the raw data, $y(t)$ is the t th variable in the generated data, $f(\cdot)$ is the transform relation. To enhance the frequency features of raw instances, the FIR filter-based approach is adopted as the transform relation.

3.2 FIR filter-based approach

The FIR filter refers to a linear shift-invariant (LSI) system [60], and the discrete input–output relation could be represented as

$$y(t) = \alpha[1]x(t) + \alpha[2]x(t-1) + \dots + \alpha[K]x(t-K+1), \quad (3)$$

where K is the order of FIR filter, $\alpha[k]$ is the k th parameter of filter, $y(t)$ and $x(t)$ are the output variable and raw data at time t , respectively. It is noteworthy that the order of filter K is usually odd to facilitate the design procedure [61].

The frequency characteristics could be designed by setting the filter parameters $\alpha[k]$, which could be generally computed by the product of impulse response $h[k]$ and window function $v[k]$ as

$$\alpha[k] = h[k]v[k]. \quad (4)$$

The impulse response $h[k]$ of four-type filter system, i.e., low pass, high pass, band pass, and band stop, could be described as:

$$h_{lp}[k] = \frac{\sin(\omega_l(k-\tau))}{\pi(k-\tau)}, \quad (5)$$

Table 1 Information of raw data

Raw data	Dimension	Data Length	Partly characteristics
Stock index	1	10567	Non-stationary
Wind speed	2	15000	Volatility
Pandemic confirmed	4	669	Short, correlation
Lorenz system	3	15000	Chaos

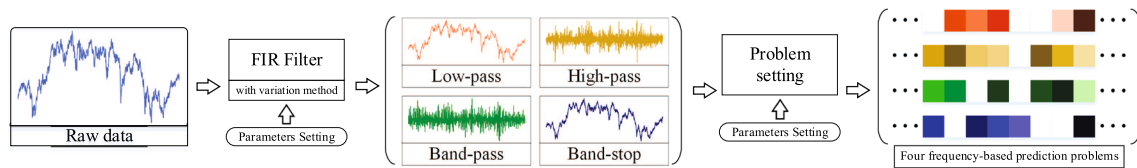


Fig. 1 Generation process of frequency-based prediction problems

$$y(t) = f \left(\begin{matrix} \text{Transform} \\ \text{relation} \end{matrix} \left[\begin{matrix} \dots & \text{red} & \text{red} & \text{red} & \text{red} & \text{red} & \dots \end{matrix} \right] x \right)$$

Fig. 2 The general data transform form

$$h_{lp}[k] = \frac{\sin(k - \tau) - \sin(\omega_l(k - \tau))}{\pi(k - \tau)}, \quad (6)$$

$$h_{bp}[k] = \frac{\sin(\omega_l(k - \tau)) - \sin(\omega_h(k - \tau))}{\pi(k - \tau)}, \quad (7)$$

$$h_{bs}[k] = \frac{\sin(k - \tau) + \sin(\omega_l(k - \tau)) - \sin(\omega_h(k - \tau))}{\pi(k - \tau)}, \quad (8)$$

where $h_{lp}[k]$, $h_{hp}[k]$, $h_{bp}[k]$, and $h_{bs}[k]$ are the impulse responses of low-pass, high-pass, band-pass, and band-stop filters at point k , respectively. ω_l and ω_h are low and high cutoff frequencies, and $\tau = \frac{K-1}{2}$ is group delay parameter.

In order to only maintain the characteristics in setting frequency, the adopted window function should have small leakage errors. The Hanning window, one of the most popular window functions [62, 63], has a narrow out-of-band spectrum [64], which could enhance the frequency characteristics of data. As a result, the Hanning function is adopted as the window function of FIR filter in this study, expressed as

$$v[k] = 0.5 \left(1 - \cos\left(\frac{k\pi}{\tau}\right) \right). \quad (9)$$

To further enhance heterogeneity of transformation, a variation method is proposed to adjust the parameters of filter. The sine function is applied to fluctuate cutoff frequencies in order to generate heterogeneous filter parameters. The fluctuating cutoff frequencies at time t could be expressed as

$$\begin{cases} \omega_{l, \text{adj}}[t] = \frac{\omega_l}{B_{l,1}} + \frac{\omega_l}{B_{l,2}} \sin\left(\frac{2\pi t}{T_l}\right) \\ \omega_{h, \text{adj}}[t] = \frac{\omega_h}{B_{h,1}} + \frac{\omega_h}{B_{h,2}} \sin\left(\frac{2\pi t}{T_h}\right) \end{cases} \quad (10)$$

where $\omega_l, \text{adj}[t]$ and $\omega_h, \text{adj}[t]$ are the varying low and high cutoff frequencies at time t , $B_{l,1}$ and $B_{l,2}$ are adjustable parameters for low cutoff frequency, $B_{h,1}$ and $B_{h,2}$ are adjustable parameters for high cutoff frequency, T_l and T_h are the variation period of low and high cutoff frequencies.

For band-pass and band-stop filters, the parameters of variation method must satisfy limit, i.e., $\omega_h, \text{adj}[k] > \omega_l, \text{adj}[k]$. The constraint could be written as

$$\omega_l \left(\frac{1}{B_{l,1}} + \frac{1}{B_{l,2}} \right) < \omega_h \left(\frac{1}{B_{h,1}} + \frac{1}{B_{h,2}} \right) \quad (11)$$

3.3 Computation simplification

To reduce the computational cost, the transformation process is divided into sub-matrices. Assume the raw data $D \in \mathbb{R}_{M \times N}$ has N dimensions and M length. The n th dimensional vector $D_n \in \mathbb{R}_{M \times 1}$ in D could be denoted as

$$D_n = [d_n(1), d_n(2), \dots, d_n(M)]^T \quad (12)$$

where $d_n(m)$ is the variable at time m in D_n . The transformed output by the K order FIR filter can be expressed as

$$u_n(m) = \alpha_{n,m}[1]d_n(m) + \alpha_{n,m}[2]d_n(m-1) + \dots + \alpha_{n,m}[K]d_n(m-K+1), \quad (13)$$

where $u_n(m)$ is the transformed variable at time m , $\alpha_{n,m}[k]$ is the k th filter parameter for $u_n(m)$. It is worth noting that FIR filter requires sufficient historical data, so m should satisfy $m \geq K$.

As a result, the transform process could be denoted as

$$U_n = P_n^T D_n, \quad (14)$$

where $U_n \in \mathbb{R}_{(M-K+1) \times 1}$ is the transformed result. P_n is the parameter matrix for n th dimensional vector D_n , including filter parameters of each transformation, could be expressed as (15), where $m' = M - K + 1$.

$$P_n = \begin{bmatrix} \alpha_{n,1}[1] & \alpha_{n,1}[2] & \cdots & \alpha_{n,1}[K] & 0 & \cdots \\ 0 & \alpha_{n,2}[1] & \alpha_{n,2}[2] & \cdots & \alpha_{n,2}[K] & 0 & \cdots \\ & & & \vdots & & & \\ 0 & \cdots & 0 & \alpha_{n,m'}[1] & \cdots & \alpha_{n,m'}[K-1] & \alpha_{n,m'}[K] \end{bmatrix}_{M \times (M-K+1)} \quad (15)$$

The proposed approach has to compute the filter parameters for each data, and the computation cost is expensive. Therefore, to reduce the computation time, the parameter matrix P'_n could be congregated by several sub-matrices, expressed as

$$P'_n = \begin{bmatrix} P'_n(1) & & & \\ & P'_n(2) & & \\ & & \ddots & \\ & & & P'_n(Q) \end{bmatrix}_{QM' \times QN'} \quad (16)$$

where $P'_n(q) \in \mathbb{R}_{M' \times N'}$ is the q th sub-matrix in P'_n , and $N' = M' + K - 1$. Q is the amount of sub-matrices, and the composition form of elements in $P'_n(q)$ is the same as (15). In this competition, for simplicity, each sub-matrices are set as the same. The transform process could be denoted as

$$U'_n = P'_n D'_n \quad (17)$$

where $U'_n \in \mathbb{R}_{(QM') \times 1}$ is the result after simplified transformation and $D'_n \in \mathbb{R}_{(QN') \times 1}$ is a part data of D_n .

3.4 Problem setting

In the consideration of various requirements in time-series tasks, the problem setting module is proposed. Through setting parameters, certain variables in transformed data are removed as blank for predicting.

Assume there are two different blank setting stages, where M_{front} is the length of preserved data, $M_{\text{stage},1}$ and $M_{\text{stage},2}$ are two setting stages range, $T_{\text{stage},1}$ and $T_{\text{stage},2}$ are computation length of two stages, $K_{\text{stage},1}$ and $K_{\text{stage},2}$ are the length of data set as blank in two period, MOD is the remainder symbol.

The main steps of problem setting module could be described as follows.

step1: Leave M_{front} length front data unprocessed, and let $s = 1$.

step2: In stage s , set $K_{\text{stage},s}$ number of continuous blanks with $T_{\text{stage},s}$ period for data $d_n(m)$, where $M_{\text{stage},(s-1)} \leq m \leq M_{\text{stage},s}$. If $s = 1$, $M_{\text{stage},(s-1)}$ is the length of preserved data M_{front} .

step3: If s is not the last setting stage, let $s = s + 1$.

Table 2 Information of problems

Problem number	Dimension	Length	Frequency characteristics	Raw data	Amount of blanks in the first dimension
I	1	7500	Low-pass	Stock index	1070
II	2	13200	Low-pass	Wind speed	4000
III	3	7500	Low-pass	Lorenz system	1900
IV	4	644	Low-pass	Pandemic confirmed	214
V	1	7500	High-pass	Stock index	888
VI	2	13200	High-pass	Wind speed	4000
VII	3	7500	High-pass	Lorenz system	1900
VIII	4	644	High-pass	Pandemic confirmed	214
IX	1	7500	Band-pass	Stock index	993
X	2	13200	Band-pass	Wind speed	4000
XI	3	7500	Band-pass	Lorenz system	1900
XII	4	644	Band-pass	Pandemic confirmed	214
XIII	1	7500	Band-stop	Stock index	875
XIV	2	13200	Band-stop	Wind speed	4000
XV	3	7500	Band-stop	Lorenz system	1900
XVI	4	644	Band-stop	Pandemic confirmed	214

The detailed problem setting procedure is described in Algorithm 1. Note that in general a certain length of last part data would be preserved for researchers to validate their approaches.

Algorithm 1 The procedure of problem setting

```

1: for  $m \leftarrow 1$  to  $M$ 
2:   if  $m \geq M_{front}$ 
3:     if  $M_{front} \leq m \leq M_{stage,1}$ 
4:       if  $m \bmod T_{stage,1} \leq K_{stage,1}$ 
5:          $d_n(m) \leftarrow \text{blank}$ 
6:       if  $M_{stage,1} \leq m \leq M_{stage,2}$ 
7:         if  $m \bmod T_{stage,2} \leq K_{stage,2}$ 
8:            $d_n(m) \leftarrow \text{blank}$ 
9:   end for

```

3.5 Computational complexity analysis

The computational complexity of NCAA2022 dataset generation process is analyzed in this subsection. According to (16), the computational complexity of P'_n parameter generation could be expressed as

$$C_{\text{para},n} = O(QM'K), n = 1, 2, \dots, N. \quad (18)$$

After calculating the parameter matrix, the computational complexity of the transformation in n th dimension using matrix multiplication, i.e., (17), could be expressed as

$$C_{\text{mul},n} = O(QN'M'), n = 1, 2, \dots, N. \quad (19)$$

So the overall computational complexity of transformation process could be expressed as

$$\begin{aligned}
C_{\text{trans}} &= \sum_{n=1}^N C_{\text{para},n} + \sum_{n=1}^N C_{\text{mul},n} \\
&= O(QM'KN) + O(QN'M'N) \\
&\approx O(QN'M'N).
\end{aligned} \quad (20)$$

Assuming that there are two different blank setting stages. For the problem setting module, according to Algorithm 1, the computational complexity could be expressed as

$$C_{\text{problem}} = O(QM'N). \quad (21)$$

4 Dataset details and evaluation

4.1 The setting parameters of NCAA2022 dataset

Four raw data are generated into 16 problems by the process mentioned in Sect. 3. Part of the generated data details is shown in Table 2. In the NCAA2022 time-series competition, the prediction task is to impute blanks in the first

Table 3 Parameters of the first dimension problem setting

Problem number	M_{front}	Stage 1	Stage 2	Stage 3
I	1000	$M_{\text{stage},1} : 3000, T_{\text{stage},1} : 100, K_{\text{stage},1} : 10$	$M_{\text{stage},2} : 5000, T_{\text{stage},2} : 100, K_{\text{stage},2} : 15$	$M_{\text{stage},3} : 7500, T_{\text{stage},3} : 130, K_{\text{stage},3} : 30$
V	1050	$M_{\text{stage},1} : 2700, T_{\text{stage},1} : 70, K_{\text{stage},1} : 7$	$M_{\text{stage},2} : 7500, T_{\text{stage},2} : 160, K_{\text{stage},2} : 24$	
IX	1000	$M_{\text{stage},1} : 3000, T_{\text{stage},1} : 100, K_{\text{stage},1} : 7$	$M_{\text{stage},2} : 5000, T_{\text{stage},2} : 100, K_{\text{stage},2} : 17$	$M_{\text{stage},3} : 7500, T_{\text{stage},3} : 130, K_{\text{stage},3} : 27$
XIII	1000	$M_{\text{stage},1} : 3200, T_{\text{stage},1} : 120, K_{\text{stage},1} : 5$	$M_{\text{stage},2} : 5000, T_{\text{stage},2} : 100, K_{\text{stage},2} : 15$	$M_{\text{stage},3} : 7500, T_{\text{stage},3} : 130, K_{\text{stage},3} : 25$
II, VI, X, and XIV	1050	$M_{\text{stage},1} : 6500, T_{\text{stage},1} : 70, K_{\text{stage},1} : 16$	$M_{\text{stage},2} : 12500, T_{\text{stage},2} : 70, K_{\text{stage},2} : 32$	
III, VII, XI, and XV	1015	$M_{\text{stage},1} : 6676, T_{\text{stage},1} : 60, K_{\text{stage},1} : 20$		
IV, VIII, XII, and XVI	100	$M_{\text{stage},1} : 300, T_{\text{stage},1} : 20, K_{\text{stage},1} : 7$	$M_{\text{stage},2} : 600, T_{\text{stage},2} : 40, K_{\text{stage},2} : 18$	

Table 4 Transformation parameters

Problem number	Sub-matrices size	Sub-matrices amount	Filter order	Cutoff frequency	Adjustable parameters	Volatile period
I	50×80	150	31	Dim1: W_l : 0.6	$B_{l,1}$: 1.5; $B_{l,2}$: 4	T_l : 17
II	60×94	220	35	Dim1: W_l : 0.3	$B_{l,1}$: 2; $B_{l,2}$: 4	T_l : 17
III	50×74	150	25	Dim2: W_l : 0.5	$B_{l,1}$: 1.5; $B_{l,2}$: 4	T_l : 17
				Dim1: W_l : 0.5	$B_{l,1}$: 2; $B_{l,2}$: 4	T_l : 17
				Dim2: W_l : 0.2	$B_{l,1}$: 3; $B_{l,2}$: 6	T_l : 13
				Dim3: W_l : 0.3	$B_{l,1}$: 1.5; $B_{l,2}$: 5	T_l : 19
IV	23×37	28	15	Dim1: W_l : 0.55	$B_{l,1}$: 2; $B_{l,2}$: 4.5	T_l : 17
				Dim2: W_l : 0.4	$B_{l,1}$: 3; $B_{l,2}$: 6	T_l : 13
				Dim3: W_l : 0.5	$B_{l,1}$: 1.5; $B_{l,2}$: 5	T_l : 9
				Dim4: W_l : 0.6	$B_{l,1}$: 1; $B_{l,2}$: 4	T_l : 11
V	50×80	150	31	Dim1: W_h : 0.5	$B_{h,1}$: 1.5; $B_{h,2}$: 3	T_h : 14
VI	60×94	220	35	Dim1: W_h : 0.6	$B_{l,1}$: 2; $B_{l,2}$: 4	T_l : 17
				Dim2: W_l : 0.5	$B_{l,1}$: 1.5; $B_{l,2}$: 3	T_l : 12
VII	50×74	150	25	Dim1: W_h : 0.5	$B_{h,1}$: 1.2; $B_{h,2}$: 5.5	T_h : 15
				Dim2: W_h : 0.4	$B_{h,1}$: 2; $B_{h,2}$: 6	T_h : 17
				Dim3: W_h : 0.65	$B_{h,1}$: 2; $B_{h,2}$: 5	T_h : 11
VIII	23×37	28	15	Dim1: W_h : 0.65	$B_{h,1}$: 1.2; $B_{h,2}$: 5	T_h : 15
				Dim2: W_h : 0.55	$B_{h,1}$: 2; $B_{h,2}$: 6	T_h : 17
				Dim3: W_h : 0.7	$B_{h,1}$: 2; $B_{h,2}$: 5	T_h : 11
				Dim4: W_h : 0.6	$B_{h,1}$: 1.3; $B_{h,2}$: 4	T_h : 5
IX	50×80	150	31	Dim1: W_l : 0.2; W_h : 0.7	$B_{l,1}$: 1; $B_{l,2}$: 5; $B_{h,1}$: 1.5; $B_{h,2}$: 3	T_l : 14; T_h : 10
X	60×94	220	35	Dim1: W_l : 0.4; W_h : 0.75	$B_{l,1}$: 2; $B_{l,2}$: 4; $B_{h,1}$: 1.5; $B_{h,2}$: 3	T_l : 17; T_h : 16
				Dim2: W_l : 0.3; W_h : 0.6	$B_{l,1}$: 2; $B_{l,2}$: 4; $B_{h,1}$: 1.5; $B_{h,2}$: 3	T_l : 17; T_h : 16
				Dim1: W_l : 0.3; W_h : 0.5	$B_{l,1}$: 2; $B_{l,2}$: 5; $B_{h,1}$: 1.2; $B_{h,2}$: 5.5	T_l : 17; T_h : 15
XI	50×74	150	25	Dim2: W_l : 0.3; W_h : 0.7	$B_{l,1}$: 2; $B_{l,2}$: 6; $B_{h,1}$: 2; $B_{h,2}$: 6	T_l : 13; T_h : 17
				Dim3: W_l : 0.3; W_h : 0.65	$B_{l,1}$: 1.5; $B_{l,2}$: 5; $B_{h,1}$: 2; $B_{h,2}$: 5	T_l : 9; T_h : 11
				Dim1: W_l : 0.2; W_h : 0.4	$B_{l,1}$: 2; $B_{l,2}$: 4.5; $B_{h,1}$: 1.2; $B_{h,2}$: 5	T_l : 17; T_h : 15
XII	23×37	28	15	Dim2: W_l : 0.1; W_h : 0.44	$B_{l,1}$: 3; $B_{l,2}$: 6; $B_{h,1}$: 2; $B_{h,2}$: 6	T_l : 13; T_h : 17
				Dim3: W_l : 0.2; W_h : 0.5	$B_{l,1}$: 1.5; $B_{l,2}$: 5; $B_{h,1}$: 2; $B_{h,2}$: 5	T_l : 9; T_h : 11
				Dim4: W_l : 0.25; W_h : 0.45	$B_{l,1}$: 1; $B_{l,2}$: 4; $B_{h,1}$: 1.3; $B_{h,2}$: 4	T_l : 11; T_h : 5
				Dim1: W_l : 0.3; W_h : 0.7	$B_{l,1}$: 2; $B_{l,2}$: 7; $B_{h,1}$: 1.5; $B_{h,2}$: 8	T_l : 17; T_h : 8
XIII	50×80	150	31	Dim1: W_l : 0.5; W_h : 0.8	$B_{l,1}$: 2; $B_{l,2}$: 4; $B_{h,1}$: 1.5; $B_{h,2}$: 3	T_l : 17; T_h : 11
				Dim2: W_l : 0.35; W_h : 0.7	$B_{l,1}$: 2; $B_{l,2}$: 4; $B_{h,1}$: 1.5; $B_{h,2}$: 3	T_l : 17; T_h : 11
XV	50×74	150	25	Dim1: W_l : 0.35; W_h : 0.55	$B_{l,1}$: 2; $B_{l,2}$: 4.5; $B_{h,1}$: 1.2; $B_{h,2}$: 5	T_l : 17; T_h : 15
				Dim2: W_l : 0.35; W_h : 0.55	$B_{l,1}$: 3; $B_{l,2}$: 6; $B_{h,1}$: 2; $B_{h,2}$: 6	T_l : 13; T_h : 17
				Dim3: W_l : 0.25; W_h : 0.55	$B_{l,1}$: 1.5; $B_{l,2}$: 5; $B_{h,1}$: 2; $B_{h,2}$: 5	T_l : 9; T_h : 11

Table 4 (continued)

Problem number	Sub-matrices size	Sub-matrices amount	Filter order	Cutoff frequency	Adjustable parameters	Volatile period
XII	23×37	28	15	Dim1: $W_l: 0.2; W_h: 0.65$	$B_{l,1}: 2; B_{l,2}: 4.5; B_{h,1}: 1.2; B_{h,2}: 5$	$T_l: 17; T_h: 15$
				Dim2: $W_l: 0.3; W_h: 0.55$	$B_{l,1}: 3; B_{l,2}: 6; B_{h,1}: 2; B_{h,2}: 6$	$T_l: 13; T_h: 17$
				Dim3: $W_l: 0.3; W_h: 0.8$	$B_{l,1}: 1.5; B_{l,2}: 5; B_{h,1}: 2; B_{h,2}: 5$	$T_l: 9; T_h: 11$
				Dim4: $W_l: 0.3; W_h: 0.75$	$B_{l,1}: 1; B_{l,2}: 4; B_{h,1}: 1.3; B_{h,2}: 4$	$T_l: 11; T_h: 5$

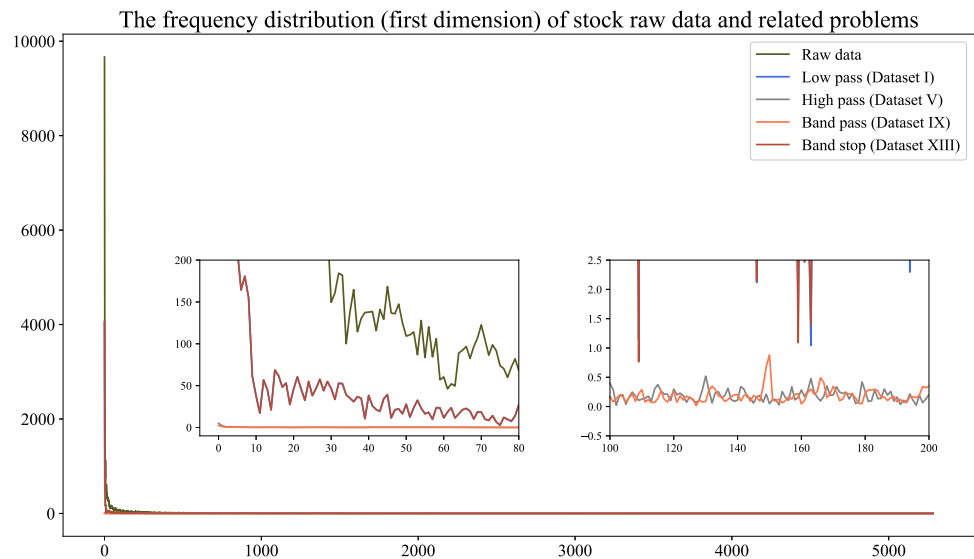
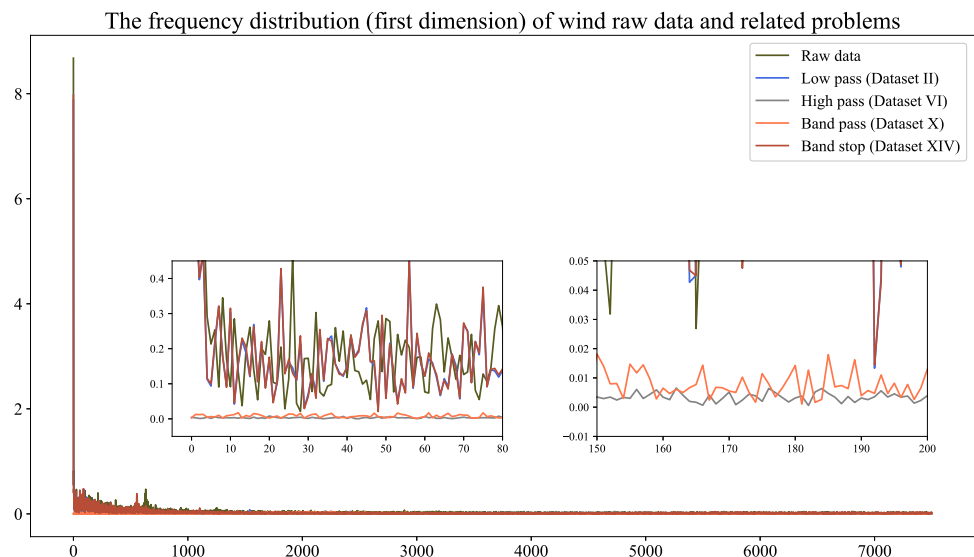
Fig. 3 The frequency distribution (first dimension) of stock raw data and related problems**Fig. 4** The frequency distribution (first dimension) of wind raw data and related problems

Fig. 5 The frequency distribution (first dimension) of chaotic raw data and related problems

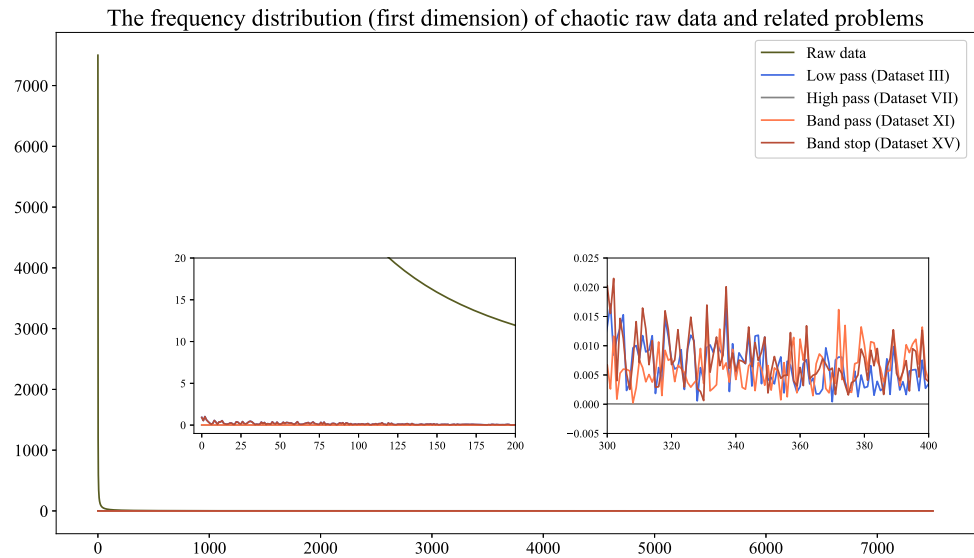
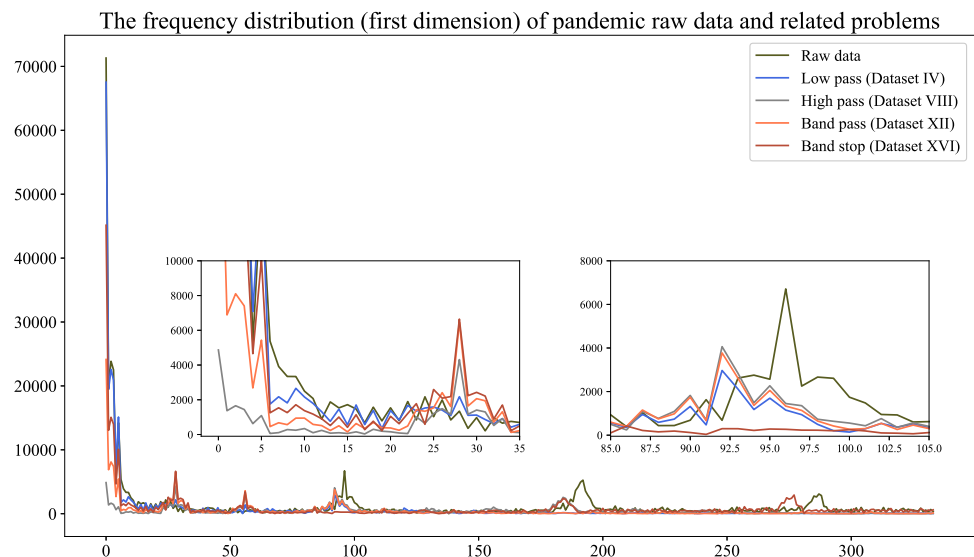


Fig. 6 The frequency distribution (first dimension) of pandemic raw data and related problems



dimension. The parameters of first dimension problem setting are shown in Table 3, and the transformation parameters are shown in Table 4.

4.2 Analysis of transformed results

To validate transformed results, discrete Fourier transform (DFT) is introduced to analyze the frequency characteristics. The continuous Fourier transform is defined as

$$F(\theta) = \int_{-\infty}^{\infty} f(t)e^{-i\theta t} dt, \quad (22)$$

where $F(\theta)$ is the spectrum of $f(t)$ at frequency θ . Equation (22) could be generated the case of discrete function by let $t \rightarrow t_k$, where t_k is discrete time [65]. The DFT could be denoted as

$$F[\theta_k] = \sum_{t_k=0}^{N-1} f[t_k]e^{-2\pi i\theta_k t_k/N} \quad (23)$$

where $F[\theta_k]$ is the interval sampling results of spectrum of series $f[t_k]$, with $t_k = 0, \dots, N-1$. The transformed result is symmetric, and only half of results contain useful information [66].

The frequency distribution characteristics of raw data and transformed results are shown in Figs. 3, 4, 5 and 6. As shown in Fig. 3, stock raw data are mainly distributed at low frequency and the transform process is effective. Comparison results from Fig. 4 reflect that wind speed raw data have a wide distribution. The chaotic system is solved by simulation, and therefore the frequency character of chaotic raw data is distributed smoothly, but the transform process would cause adding a slight noise, which could be

observed from Fig. 5. Although the pandemic-related data are short, the periodic characteristic is obvious, as shown in Fig. 6. The DFT analysis could illustrate that transform process could effectively enhance the frequency characteristics of raw data.

4.3 Evaluation metric of NCAA2022 competition

The mean absolute percentage error (MAPE) is adopted to evaluate the prediction results of different problems in this competition. The MAPE of each problem could be expressed as follows:

$$MAPE_i = \frac{1}{Q} \sum_{j=1}^Q \left| \frac{d_{ij}^{\text{actual}} - d_{ij}^{\text{predicted}}}{d_{ij}^{\text{actual}}} \right|, \quad (24)$$

where $MAPE_i$ is the MAPE of i th problem, Q is the amount of removed data in the first dimension, $d_{ij}^{\text{predicted}}$ and d_{ij}^{actual} are prediction value and actual data, respectively.

As raw data are transformed by four-type filters, the evaluation scores are divided according to the filter type processed. In other words, the prediction results are ranked into problem number 1–4, 5–8, 9–12, and 13–16, 4 groups. The ranking of each group is ordered by the performance score, denoted as

$$\text{score} = \frac{1}{4} \sum MAPE_i. \quad (25)$$

4.4 Other evaluation metrics

Besides abovementioned evaluation method, there are certain evaluation metrics are used in other researches, such as root-mean-square error (RMSE), mean absolute error (MAE), normalized root-mean-square (nRMSE), and symmetric mean absolute percentage error (sMAPE), as shown in (26)–(29).

$$MAE_i = \frac{1}{Q} \sum_{j=1}^Q |d_{ij}^{\text{actual}} - d_{ij}^{\text{predicted}}| \quad (26)$$

$$RMSE_i = \sqrt{\frac{1}{Q} \sum_{j=1}^Q (d_{ij}^{\text{actual}} - d_{ij}^{\text{predicted}})^2} \quad (27)$$

$$nRMSE_i = \frac{1}{d_i^{\text{mean}}} \sqrt{\frac{1}{Q} \sum_{j=1}^Q (d_{ij}^{\text{actual}} - d_{ij}^{\text{predicted}})^2} \quad (28)$$

$$sMAPE_i = \frac{1}{Q} \sum_{j=1}^Q \frac{|d_{ij}^{\text{actual}} - d_{ij}^{\text{predicted}}|}{|d_{ij}^{\text{actual}}| + |d_{ij}^{\text{predicted}}|} \quad (29)$$

Table 5 MAPE results of baseline methods

Problem	BP	ESN	ARX	RNN	LSTM
I	0.728	0.641	22.537	0.766	0.795
II	12.879	12.584	40.39	15.838	16.034
III	17.198	15.878	29.482	18.544	18.24
IV	9.083	5.219	28.123	5.874	35.714
V	36.007	34.77	35.963	35.806	36.248
VI	40.5	36.729	37.905	37.033	36.2
VII	32.592	34.701	42.145	32.906	33.715
VIII	42.652	40.183	43.279	40.599	42.715
IX	37.567	37.773	39.611	34.3	34.901
X	39.234	36.381	42.3	37.086	44.939
XI	36.782	40.299	42.084	38.274	35.736
XII	28.293	14.991	34.935	20.913	44.063
XIII	0.591	0.57	0.523	1.215	1.171
XIV	14.493	15.275	15.075	19.984	23.533
XV	17.261	16.33	30.892	17.701	17.48
XVI	11.814	9.575	12.281	9.372	11.302

The bold numbers in the table are the best prediction results in those problems among baseline approaches

where MAE_i , $RMSE_i$, $nRMSE_i$, and $sMAPE_i$ are MAE, RMSE, nRMSE, and sMAPE of i th problem, respectively, d_i^{mean} are the mean value of actual data.

The four indicators could reflect different characteristics of prediction performance. Generally, the MAE and sMAPE show the overall error and is adopted to evaluate wind-related and pandemic-related problems. The metrics RMSE and nRMSE are able to amplify the extreme prediction error, so they are suitable for evaluating the problems with continuity, e.g., stock-related and chaos-related problems.

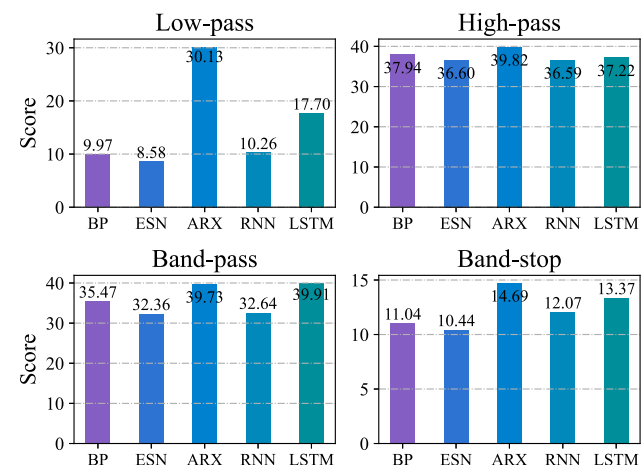


Fig. 7 The evaluation score of baseline approaches' performance

4.5 Baseline experiment

In this subsection, we select five baseline approaches, i.e., autoregressive exogenous (ARX) model, back-propagation (BP) network, echo state network (ESN), recurrent neural network (RNN), and long short-term memory (LSTM), to run on the NCAA2022 dataset.

The settings of hyperparameters are referred to [35, 67], and [68]. For BP network, 2 hidden layers with 250 nodes are included, and learning rate is chosen as 0.03. In RNN and LSTM network, 2 hidden layers, which has 200 and 250 units, are included, and learning rate is set as 0.01. The variance of Gaussian noise in ARX model is set as 0.5. The parameter selection of ESN is different from BP-based network, so we take an enumerate experiment for each hyperparameter in steps of 0.1. Then the hyperparameter with the best result is used in the baseline experiments: the reservoir size is set as 70; the uniform range of input weight and the spectral radius of reservoir are set to be 0.7 and 0.8, respectively.

For simplification and experiment fairness, the historical data in the first dimension is only considered as input for baseline approaches, and the preserved part of data is selected as the training set. The sliding window is adopted to pre-process data, and the window size of each baseline methods is set as 7.

The MAPE results of baseline approaches are shown in Table 5. Each method has better performance than others in certain problems. Due to the special training mechanism, ESN has well convergence property [69]. Therefore, ESN has better overall performance than other baseline approaches in the case of short training length, especially on stock-related problems. In addition, it is hard for LSTM to learn the hyperparameters with a short training set, thus the performance of LSTM on pandemic-related problems is worse than the RNN's. BP neural network has fewer weight parameters and the requirement of training set than RNN and LSTM networks [70], and therefore BP neural network performance is closer or better than RNN in problems with mainly low-pass data. However, due to the memory capability, in pandemic-related data, which has strong periodical characteristics, RNN performs better than BP network's performance. ARX is more sensitive to low-frequency data as its algorithm is simple, and the added Gaussian noise makes it performs well in band-stop data, even having the best performance than other neural-based approaches in problem XIII.

The scores of baseline methods are shown in Fig. 7. The characteristics of baseline approaches in problems with various frequency features could be reflected from the scores. The ARX model has a poor performance in low-pass problems; however, the ARX model has a

performance close to that of other baseline approaches in other types of problems. The performance of neural-based approaches in high-pass problems validates the conclusion in [36], i.e., neural network-based methods have poor generalization in high-frequency signals. Different baseline approaches show varying performance on each category, which indicate our proposed benchmarks could provide a frequency-based evaluation platform for time-series prediction algorithms.

5 Conclusion

In this paper, four raw data are transformed into 16 problems as the prediction task of NCAA2022 time-series competition. Each raw datum has various characteristics, and covers popular research areas. Different from existing competition datasets, from the perspective of frequency domain, four raw data are transformed into various problems. To reduce the computational burden, filter parameters are divided into sub-matrices. The comparative results in the frequency domain illustrate the transform process is effective. With the NCAA2022 evaluation metric, five baseline approaches are run on problems to further validate the benchmarking capability of NCAA2022 dataset.

Although the effectiveness of NCAA2022 dataset is validated, some limitations still exist. For example, the NCAA2022 dataset is generated from only four popular-studied instances; however, there are other hot topics of time-series prediction not included. Furthermore, the current scale of NCAA2022 may not fit those large-scale prediction approaches. Our future work is to further add various characteristics instances and research the evaluation metrics to efficiently demonstrate the frequency features of algorithms.

Funding This work is supported by the National Key Research and Development Program of China (No. 2022YFE0198900, No. 2021YFF0500903), the National Natural Science Foundation of China (No. 52178271, No. 52077213), and the Fundamental Research Funds for the Central Universities (No. 2022CDJKYJH034).

Data availability Datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

- Mahalakshmi G, Sridevi S, Rajaram S (2016) A survey on forecasting of time series data. In: 2016 International conference on computing technologies and intelligent data engineering (ICCTIDE'16). IEEE, pp 1–8
- Taieb SB, Bontempi G, Atiya AF, Sorjamaa A (2012) A review and comparison of strategies for multi-step ahead time series forecasting based on the nn5 forecasting competition. *Expert Syst Appl* 39(8):7067–7083
- Thung K-H, Wee C-Y (2018) A brief review on multi-task learning. *Multimed Tools Appl* 77(22):29705–29725
- Fan J, Zhang K, Huang Y, Zhu Y, Chen B (2021) Parallel spatio-temporal attention-based tcn for multivariate time series prediction. *Neural Comput Appl* 1–10
- Lin W-C, Tsai C-F (2020) Missing value imputation: a review and analysis of the literature (2006–2017). *Artif Intell Rev* 53(2):1487–1509
- Gan M, Cheng Y, Liu K, Zhang G-I (2014) Seasonal and trend time series forecasting based on a quasi-linear autoregressive model. *Appl Soft Comput* 24:13–18
- Mena R, Rodríguez F, Castilla M, Arahall MR (2014) A prediction model based on neural networks for the energy consumption of a bioclimatic building. *Energy Build* 82:142–155
- Jain RK, Smith KM, Culligan PJ, Taylor JE (2014) Forecasting energy consumption of multi-family residential buildings using support vector regression: investigating the impact of temporal and spatial monitoring granularity on performance accuracy. *Appl Energy* 123:168–178
- Boyacioglu MA, Avci D (2010) An adaptive network-based fuzzy inference system (anfis) for the prediction of stock market return: the case of the Istanbul stock exchange. *Expert Syst Appl* 37(12):7908–7912
- Adhikari R, Agrawal R (2014) A combination of artificial neural network and random walk models for financial time series forecasting. *Neural Comput Appl* 24(6):1441–1449
- Livieris IE, Pintelas E, Pintelas P (2020) A cnn-lstm model for gold price time-series forecasting. *Neural Comput Appl* 32:17351–17360
- de Araújo AR, Ferreira TAE (2013) A morphological-rank-linear evolutionary method for stock market prediction. *Inf Sci* 237:3–17
- Dong L, Zhang H, Yang K, Zhou D, Shi J, Ma J (2022) Crowd Counting by Using Top-k Relations: A Mixed Ground-Truth CNN Framework. *IEEE Trans Consum Electron* 68(3):307–316. <https://doi.org/10.1109/TCE.2022.3190384>.
- Liu H, Chen C (2019) Data processing strategies in wind energy forecasting models and applications: a comprehensive review. *Appl Energy* 249:392–408
- Yang B, Zhong L, Wang J, Shu H, Zhang X, Yu T, Sun L (2021) State-of-the-art one-stop handbook on wind forecasting technologies: an overview of classifications, methodologies, and analysis. *J Clean Prod* 283:124628
- Gómez D, Rojas A (2016) An empirical overview of the no free lunch theorem and its effect on real-world machine learning classification. *Neural Comput* 28(1):216–228
- Makridakis S, Hibon M (2000) The m3-competition: results, conclusions and implications. *Int J Forecast* 16(4):451–476
- Koning AJ, Franses PH, Hibon M, Stekler HO (2005) The m3 competition: statistical tests of the results. *Int J Forecast* 21(3):397–409
- Makridakis S, Spiliotis E, Assimakopoulos V (2020) The m4 competition: 100,000 time series and 61 forecasting methods. *Int J Forecast* 36(1):54–74
- Crone SF, Hibon M, Nikolopoulos K (2011) Advances in forecasting with neural networks? empirical evidence from the nn3 competition on time series prediction. *Int J Forecast* 27(3):635–660
- Crone S (2008) Results of the nn5 time series forecasting competition. In: WCCI 2008, IEEE World Congress on Computational Intelligence
- Godahehwa R, Bergmeir C, Webb GI, Hyndman RJ, Montero-Manso P (2021) Monash time series forecasting archive. arXiv preprint [arXiv:2105.06643](https://arxiv.org/abs/2105.06643)
- Abbasimehr H, Paki R, Bahrini A (2022) A novel approach based on combining deep learning models with statistical methods for covid-19 time series forecasting. *Neural Comput Appl* 1–15
- Dong E, Du H, Gardner L (2020) An interactive web-based dashboard to track covid-19 in real time. *Lancet Infect Dis* 20(5):533–534
- Narayan PK, Iyke BN, Sharma S (2021) New measures of the covid-19 pandemic: a new time-series dataset
- Piccoli L, Dzankic J, Ruedin D (2021) Citizenship, migration and mobility in a pandemic (cmmpp): a global dataset of covid-19 restrictions on human movement. *PloS One* 16(3):0248066
- Li H, Wang Z, Hong T (2021) A synthetic building operation dataset. *Sci Data* 8(1):1–13
- Pullinger M, Kilgour J, Goddard N, Berliner N, Webb L, Dziukovska M, Lovell H, Mann J, Sutton C, Webb J et al (2021) The ideal household energy dataset, electricity, gas, contextual sensor data and survey data for 255 uk homes. *Sci Data* 8(1):1–18
- Chavat J, Nesmachnow S, Graneri J, Alvez G (2022) Ecd-uy, detailed household electricity consumption dataset of uruguay. *Sci Data* 9(1):1–16
- Zheng X, Xu N, Trinh L, Wu D, Huang T, Sivaranjani S, Liu Y, Xie L (2022) A multi-scale time-series dataset with benchmark for machine learning in decarbonized energy grids. *Sci Data* 9(1):1–18
- Angrzyk RA, Martens PC, Aydin B, Kempton D, Mahajan SS, Basodi S, Ahmadzadeh A, Cai X, Filali Boubrahimi S, Hamdi SM et al (2020) Multivariate time series dataset for space weather data analytics. *Sci data* 7(1):1–13
- Kierdorf J, Junker-Frohn LV, Delaney M, Olave MD, Burkart A, Jaenicke H, Muller O, Rascher U, Roscher R (2022) Growli-flower: an image time-series dataset for growth analysis of cauliflower. *J Field Robot*
- Schneider M, Broszeit A, Körner M (2021) Eurocrops: a pan-european dataset for time series crop type classification. arXiv preprint [arXiv:2106.08151](https://arxiv.org/abs/2106.08151)
- Sailer G, Eichermüller J, Poetsch J, Paczkowski S, Pelz S, Oechsner H, Müller J (2021) Characterization of the separately collected organic fraction of municipal solid waste (ofmsw) from rural and urban districts for a one-year period in germany. *Waste Manag* 131:471–482
- Han Z, Zhao J, Leung H, Ma KF, Wang W (2019) A review of deep learning models for time series prediction. *IEEE Sens J* 21(6):7833–7848
- Xu Z-QJ, Zhang Y, Luo T, Xiao Y, Ma Z (2019) Frequency principle: Fourier analysis sheds light on deep neural networks. arXiv preprint [arXiv:1901.06523](https://arxiv.org/abs/1901.06523)
- Cai W, Li X, Liu L (2019) Phasednn-a parallel phase shift deep neural network for adaptive wideband learning. arXiv preprint [arXiv:1905.01389](https://arxiv.org/abs/1905.01389)
- Kim T, King BR (2020) Time series prediction using deep echo state networks. *Neural Comput Appl* 32(23):17769–17787
- Bilgili M, İlhan A, Ünal Ş (2022) Time-series prediction of hourly atmospheric pressure using anfis and lstm approaches. *Neural Comput Appl* 1–16

40. Wu B, Wang L, Tao R, Zeng Y-R (2022) Interpretable tourism volume forecasting with multivariate time series under the impact of covid-19. *Neural Comput Appl* 1–27
41. Zouaidia K, Rais MS, Ghanemi S (2023) Weather forecasting based on hybrid decomposition methods and adaptive deep learning strategy. *Neural Comput Appl* 1–16
42. Our-World-in-Data: Covid-19 Data. Accessed 26 Nov 2021, [Online]. Available: <https://github.com/owid/covid-19-data/tree/master/public/data>
43. Investing.com: Dow Jones Industrial Average. Accessed 26 Nov 2021, [Online]. Available: <https://cn.investing.com/indices/us-30-historical-data>
44. IOWA-State-University: IOWA-State-University, Wind Data. Accessed 26 Nov 2021, [Online]. Available: <https://mesonet.agron.iastate.edu/request/download.phtml>
45. Yang C, Qiao J, Ahmad Z, Nie K, Wang L (2019) Online sequential echo state network with sparse rls algorithm for time series prediction. *Neural Netw* 118:32–42
46. Cheng S, Qiu M (2022) Observation error covariance specification in dynamical systems for data assimilation using recurrent neural networks. *Neural Comput Appl* 34(16):13149–13167
47. Henrique BM, Sobreiro VA, Kimura H (2019) Literature review: machine learning techniques applied to financial market prediction. *Expert Syst Appl* 124:226–251
48. Patra A, Das S, Mishra S, Senapati MR (2017) An adaptive local linear optimized radial basis functional neural network model for financial time series prediction. *Neural Comput Appl* 28(1):101–110
49. Atsalakis GS, Valavanis KP (2009) Surveying stock market forecasting techniques-part ii: soft computing methods. *Expert Syst Appl* 36(3):5932–5941
50. Kumar R, Kumar P, Kumar Y (2022) Three stage fusion for effective time series forecasting using bi-lstm-arma and improved de-abc algorithm. *Neural Comput Appl* 34(21):18421–18437
51. Colak I, Sagioglu S, Yesilbudak M (2012) Data mining and wind power prediction: a literature review. *Renew Energy* 46:241–247
52. Abdoos AA (2016) A new intelligent method based on combination of vmd and elm for short term wind power forecasting. *Neurocomputing* 203:111–120
53. Pan T, Sumalee A, Zhong R-X, Indra-Payoong N (2013) Short-term traffic state prediction based on temporal-spatial correlation. *IEEE Trans Intell Transp Syst* 14(3):1242–1254
54. Gautam Y (2021) Transfer learning for covid-19 cases and deaths forecast using lstm network. *ISA Trans*
55. Yan H, Zhang H, Shi J, Ma J, Xu X (2023) Inspiration Transfer for Intelligent Design: A Generative Adversarial Network with Fashion Attributes Disentanglement. *IEEE Trans on Consum Electron* 1–1. <https://doi.org/10.1109/TCE.2023.3255831>.
56. Ghosal S, Bhattacharyya R, Majumder M (2020) Impact of complete lockdown on total infection and death rates: a hierarchical cluster analysis. *Diabetes Metabol Syndr Clin Res Rev* 14(4):707–711
57. Dubois P, Gomez T, Planckaert L, Perret L (2020) Data-driven predictions of the lorenz system. *Phys D Nonlinear Phenom* 408:132495
58. Wang L, Zou F, Hei X, Yang D, Chen D, Jiang Q, Cao Z (2014) A hybridization of teaching-learning-based optimization and differential evolution for chaotic time series prediction. *Neural Comput Appl* 25(6):1407–1422
59. Morales L (2021) Python and Physics: Lorenz and Rossler Systems. Accessed 26 Nov, [Online]. Available: <https://medium.com/codex/python-and-physics-lorenz-and-rossler-systems-65735791f5a2>
60. Cetin AE, Gerek ON, Yardimci Y (1997) Equiripple fir filter design by the fft algorithm. *IEEE Signal Process Mag* 14(2):60–64
61. Lim Y, Parker S (1983) Fir filter design over a discrete powers-of-two coefficient space. *IEEE Trans Acoust Speech Signal Process* 31(3):583–591
62. Zafar M, Awais MN, Shehzad MN (2022) Computationally efficient memristor model based on hann window function. *Microelectron J* 105476
63. Wieseman CD, Mukhopadhyay V, Hoadley ST, Pototzky AS (1995) Techniques in on-line performance evaluation of multi-loop digital control systems and their application. In: Leondes CT (ed) *Discrete-Time Control System Analysis and Design*, vol 71. Control and Dynamic Systems. Academic Press, Massachusetts, pp 263–289
64. Wilson S, Dobre O (2016) Multicarrier transmission in a frequency-selective channel. In: Academic press library in mobile and wireless communications. Elsevier, Amsterdam, pp 333–367
65. Jin L, Liang L (2010) A power-of-two fft algorithm and structure for drm receiver. *IEEE Trans Consum Electron* 56(4):2061–2066
66. Wiedemann U, Ommer D, Muschallik C (2002) Automatic measurements of digital receivers using sampled signals. *IEEE Trans Consum Electron* 48(3):539–547. <https://doi.org/10.1109/TCE.2002.1037039>
67. Parmezan ARS, Souza VM, Batista GE (2019) Evaluation of statistical and machine learning models for time series prediction: identifying the state-of-the-art and the best conditions for the use of each model. *Inf Sci* 484:302–337
68. Li Q, Wu Z, Zhang H (2020) Spatio-temporal modeling with enhanced flexibility and robustness of solar irradiance prediction: a chain-structure echo state network approach. *J Clean Prod* 261:121151
69. Dai J, Zhang P, Mazumdar J, Harley RG, Venayagamoorthy G (2008) A comparison of mlp, rnn and esn in determining harmonic contributions from nonlinear loads. In: 2008 34th Annual Conference of IEEE Industrial Electronics. IEEE, pp 3025–3032
70. Graves A, Schmidhuber J (2005) Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Netw* 18(5–6):602–610

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.