IEEE Workshop on Nonlinear Circuit Networks
December 4-5, 2020
NCN 2020

Chaotic Time Series Analysis by Neural Network Using Features of Reconstructed Attractors with Isometric Mapping Method

Tomoya Takata, Yoko Uwate and Yoshifumi Nishio
Dept. of Electrical and Electronic Engineering, Tokushima University
2-1 Minami-Josanjima, Tokushima 770–8506, Japan
Email:{tomoya, uwate, nishio}@ee.tokushima-u.ac.jp

***Abstract*—In recent years, big data analysis using Neural Networks has attracted attention. One of the effective applications is the analysis of time series data with irregular vibrations. Among them, it is difficult to analyze the data using Neural Networks (NN). Therefore, it is important to search for the effective features of the data. In this study, we transform the number of dimensions of the data and refer to features suitable for NN classification. We investigate which dimensional features are suitable for classification.**

I. Introduction

The researches on time series data analysis have been actively conducted in recent years. This data has various characteristics, such as oscillating regularly and irregularly. Chaos theory mainly handles irregularly oscillating data. The theory deals with a phenomenon that appears in a part of a dynamical system. It shows a complicated state that cannot be predicted due to numerical errors. Unpredictability is not random and follows deterministic rules. However, the methods for observing other phenomena are required because it cannot be obtained by the integration method. One of them is the time delay coordinate system which is a kind of chaos theory [1]. 1-dimensional data (1d-data) can be converted into multidimensional data called attractors by using this method. Many phenomena in the real world are indicated by the attractor and are described by differential and difference equations as dissipative dynamical systems with energy dissipation. In addition, the attractors are often applied in various fields such as cranial nerve system and engineering system [2].

In recent years, the researches on time series data analysis using Neural Networks (NN) such as 1-dimensional Convolutional Neural Network (1d-CNN) and Recurrent Neural Network (RNN) have been actively performed. The reason why NN attract attention is that NN can learn and recognize input information by itself. However, even if the models are used, it is difficult to analyze time series data with very irregular vibrations. It is difficult for NN to learn the numerical and time series characteristics of the data. Therefore, it is necessary for NN to preprocess the data.

In this study, the data are extended to multidimensional data using the time delay coordinate system and compressed to 1d-data using Isometric mapping (Isomap) in order to solve this problem. The objective is to find features that NN can easily learn from time series data with irregular vibrations.

II. Convolutional Neural Network

Convolutional Neural Network (CNN) is one of the NN and is mainly used in image recognition [3]. It is established in 1956 as an academic field, and it has received a lot of attention since winning the image recognition competition in 2012. Recently, various networks such as VGGNet and ResNet have been devised, and the misrecognition rate has been greatly improved [4], [5]. CNN is used in various fields such as medicine and automobiles, and its application range is expanding. It already puts into practical use in automated driving, robots, surveillance cameras and so on. Besides images such as object detection and segmentation, it has achieved tremendous performance in many tasks such as natural language processing and speech processing. CNN is inspired by the biological processes and mimics the arrangement of the visual cortex of animals. It consists of an input layer, an output layer, and a number of hidden layers between them. It is usually composed of the convolutional layers, the pooling layers, and the fully connected layers (Fig. 1).

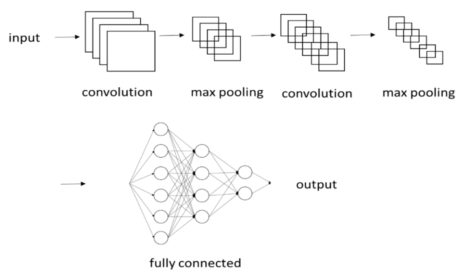

Fig. 1: An example of CNN structure.

III. Dataset

The data used in this study are the values of the x-axis of the Lorenz equation. The Lorenz system is ordinary differential equation first studied by Edward Lorenz and Ellen Fetter. It is notable for having chaotic solutions for certain parameter values and initial conditions. The Lorenz equation is described in Eq. (1). An equation for the three variables x, y and z, the behavior of the system is determined by the three constants, σ, ρ and β. In particular, the Lorenz attractor is a set of chaotic solutions of the Lorenz system (Fig. 2).

$$f(x) = \begin{cases} \dfrac{dx}{dt} = \sigma(y - x) \\ \dfrac{dy}{dt} = -\rho x - y - xz \\ \dfrac{dz}{dt} = xy - \beta z \end{cases} \tag{1}$$

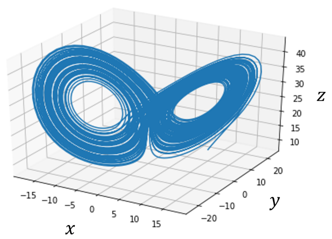

Fig. 2: The figure of the Lorenz equation.

The equation is guaranteed to be chaotic by setting $\sigma = 10$, $\beta = 8/3$ and $\rho = 28$ [6]. Chaos theory proves that the data is chaotic when the value called the Lyapunov exponent is greater than or equal to 0. Figure 3 shows the change in the Lyapunov exponent when the value of ρ is changed [7]. Therefore, in this study, the three types of the chaotic data are used with setting ρ = 28, 29 and 30.

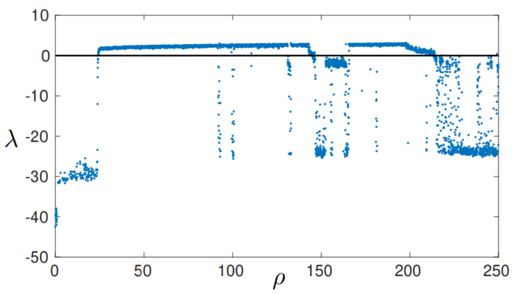

Fig. 3: Lyapunov exponents of the Lorenz equation.

IV. Proposed Method

In this study, we propose a method of feature translation for using multidimensional features of time series data. The procedure is shown below. Figure 4 shows the flow of the proposed method of this study.

1) 1d-data is converted into multidimensional data using the time delay coordinate system with the False Nearest Neighbor (FNN) method.

2) The extended data is compressed by using Isomap method.

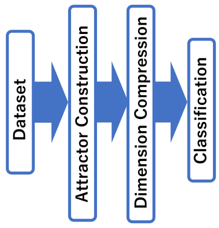

Fig. 4: Proposed method.

(a) Attractor construction

Time delay coordinate system is generally known as the method for extending dimensions. Let the value of data at a specific time be $x(n)$. Furthermore, if the time delay value is τ, this system can be expressed by Eq. (2) and showed in Fig. 5.

$$f(x) = [x(n),\ x(n+\tau),\ x(n+2\tau)\ ...] \tag{2}$$

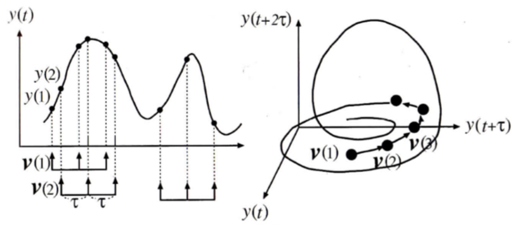

Fig. 5: Time delay coordinate system.

FNN is the most popular tool for the selection of the minimal embedding dimension. When the embedded dimension of the attractor construction is increased, the false neighbor points should disappear, and the dimension is considered to be the minimum embedded dimension. The false neighbor points will be described in Fig. 6. When a 2-dimensional attractor is converted into 3-dimensional attractor, it is called false neighbor points in that case the distance between B and C is long. Conversely, it is called true neighbor in that case the distance between C and D remains short.

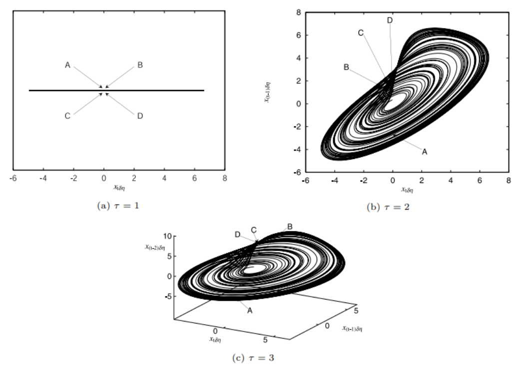

Fig. 6: The description of the true and false neighbors points.

In this study, we extend the data beyond the embedded dimension where the false neighbor points completely disappear. Figure 7 shows the generally known fraction of decreasing of the false neighbor points when the dimensions of the chaotic data are extended.

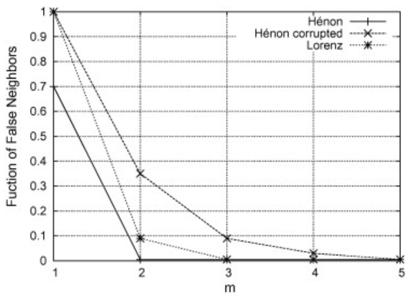

Fig. 7: The fraction of the false neighbor points of the chaotic data.

(b) Dimension Compression

Isomap is a nonlinear dimensionality reduction method used for dimension compression. It is one of several widely used low-dimensional embedding methods. Isomap is used for computing a quasi-isometric, low-dimensional embedding of a set of high-dimensional data points. The method provides the solution for estimating the intrinsic geometry of the data based on a rough estimation of each data point's neighbors on the manifold. Isomap is highly efficient and generally applicable to a broad range of data sources and dimensionalities. The procedure of the method is shown in the following steps.

1) Determine the neighbors of each point.
2) Construct a neighborhood graph.
3) Compute shortest path between two nodes.
4) Compute lower-dimensional embedding.

The keypoint of the method is that this method takes advantage of the fact that 3-dimensional manifolds can actually represented in lower dimensions. In Isomap, the geodestic distance drawed with solid lines is used instead of the euclidean distance with dashed lines to extract the characteristics of manifolds.

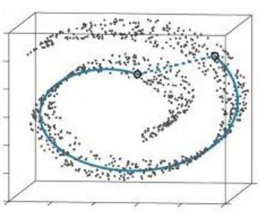

Fig. 8: The description of geodestic and euclidean distance.

V. Simulation Model

In this study, the 1-dimensional CNN (1d-CNN) is used for classification model. Figure 9 shows the structure of the 1d-CNN.

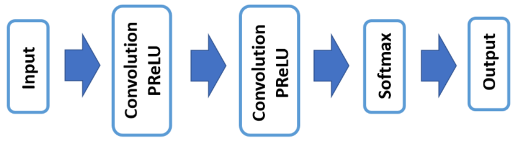

Fig. 9: 1d-CNN structure used in this study.

The 1d-CNN consists of dozens of layers, each layer learning to detect different features of 1d-data. For each learning data, multiple filters with different resolutions are applied and the output of each convolution data is used as input to the next layer. By repeating the training, the filters increase the level of complexity of the features. The convolution of 1d-CNN is expressed in Eq. (3). n is the length of the convolution filter. k is the filter number. x is the input image data. In this study, the number of the first filters is 32, and that of the second filters is 64.

$$f(x) = \sum_{t=0}^{n-1} w_t^{(k)} x_{j+t} + b^{(k)} \tag{3}$$

The pooling layer is removed for speedup. Thereupon, the activation function Parametric Rectified Linear Unit (PReLU) is put in after the convolutional layer to prevent over learning [8]. The negative gradient $a \quad (0 < a < 1)$ is adjusted by network learning. The PReLU function is expressed in Eq. (4).

$$f(x) = \begin{cases} x & (x \geq 0) \\ ax & (x < 0) \end{cases} \tag{4}$$

The classification probability is derived by the softmax function shown in Eq. (5). p is the probability of becoming class j, x is the output of the NN, and n is the total number of discriminant classes. In this study, the total number C of discriminant classes is 3. For all classification, $f(x)$ satisfies $0 < f(x) < 1$.

$$f(x) = \frac{exp(x_j)}{\sum_{i=0}^{n} exp(x_i)} \tag{5}$$

Table I shows the number of the data with changing ρ used in this study. Table II shows the batch size, epochs, and learning rate in this study. Batch size is the amount of data processed at once. The epoch is the number of the learnings. The learning rate is the degree of learning progress. By setting as shown in Table II , the training accuracies reach 100%.

TABLE I: The number of the data.

	ρ = 28	ρ = 29	ρ = 30
train data	300	300	300
test data	60	60	60

TABLE II: Learning parameters.

	Batch size	Learning rate	Epoch
1D-CNN	50	1e-3	30

VI. Simulation results

We investigate the average of 10 times of test accuracy. Table Ⅲ shows the test accuracy compressed by Isomap with changing the extended dimension. The test accuracy using the data extended to 5 dimensions and compressed by Isomap is the best. Other test accuracies of the proposed method are almost the same as the test accuracy of the conventional method. It should retain its original properties as the false neighbor points disappear when extended to 3 or more dimensions.

TABLE III: Test accuracy of the proposed method.

	test accuracy
original	0.566
3dim+Isomap	0.572
4dim+Isomap	0.572
5dim+Isomap	0.646

This result leads us to wonder why the data more accurate. Thereupon, we investigate that the relationship between test accuracy and the fraction of the false neighbor points of the data. Tables Ⅳ and Ⅴ show the fractions of the false neighbor points of the training and test data. From these tables, it can be seen that each test accuracy increases as the fraction of the false neighbor points decrease.

TABLE IV: The fraction of false neighbor points of the train data.

	$\rho = 28$	$\rho = 29$	$\rho = 30$
original	0.711	0.734	0.746
3dim+Isomap	0.711	0.702	0.711
4dim+Isomap	0.634	0.628	0.626
5dim+Isomap	0.589	0.575	0.574

TABLE V: The fraction of false neighbor points of the test data.

	$\rho = 28$	$\rho = 29$	$\rho = 30$
original	0.727	0.732	0.695
3dim+Isomap	0.695	0.695	0.695
4dim+Isomap	0.624	0.621	0.622
5dim+Isomap	0.586	0.580	0.578

VII. Conclusion

In this study, we proposed a method that uses time delay coordinate system to extend time series data into multidimensional space and compresses the data using Isomap. It was confirmed that how the accuracy of the time series classification changes using 1d-CNN by extending 1d-data to multidimensional data. As a result, some of the test accuracies using the proposed method were superior to those using the conventional method. In addition, it was found that there is the important relationship between the test accuracy and the fraction of false neighbor points of the data.

By this relationship, we will consider the method for compressing the dimension of the data with minimizing the fraction of false neighbor points. In addition, we will use other model such as RNN which is specialized in analysis of the time series flow, and use other actual measurement dataset.

References

[1] K. Aihara, T. Ikeguchi, Y. Yamada and M. Komuro, “Fundamentals and applications of chaotic time series analysis”, Industrial book, p.23, 2000.

[2] W. Gilpin, “Deep reconstruction of strange attractors from time series”, arXiv:2002.05909, Submitted on 14 Oct 2020 (v1), last revised 4 Nov 2018.

[3] M. Matsugu, K. Mori, Y. Mitari and Y. Kaneda, “Subject independent facial expression recognition with robust face detection using a convolutional neural network”, Neural Networks 16., vol.5, pp.555-559, 2003.

[4] K. Simonyan, A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, IEEE International Conference on Computer Vision, arXiv:1409.1556, 10 Apr 2015.

[5] K. He, X. Zhang, S. Ren, J. Sun, “Deep Residual Learning for Image Recognition”, IEEE International Conference on Computer Vision and Pattern Recognition, arXiv:1512.03385, 10 Dec 2015.

[6] E. D. Lorenz, “Deterministic nonperiodic flow”, J. Atmosph. Sci. 20 (1963), 130141, revised form 7 January 1963.

[7] N. Boull, V. Dallas, Y. Nakatsukasa and D. Samaddar, “Classification of chaotic time series with deep learning”, Submitted on 30 Apr 2014 (v1), last revised 3 Dec 2019 (this version, v3).

[8] K. He, X. Zhang, S. Ren and J. Sun, “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”, IEEE International Conference on Computer Vision and Pattern Recognition, arXiv:1502.01852, 6 Feb 2015.