2016 International Symposium on Nonlinear Theory and Its Applications,
NOLTA2016, Yugawara, Japan, November 27th-30th, 2016

Encoding Multi-Dimensional Time Series Data with Reservoir Computing

Yuichi Katori

The School of Systems Information Science, Future University Hakodate,
116–2 Kamedanakano-cho, Hakodate, Hokkaido, Japan,
Institute of Industrial Science, The University of Tokyo,
4–6–1 Komaba, Meguro-ku, Tokyo, Japan.
Email: katori@fun.ac.jp

Abstract—Encoding of given time series data can be basis of various information processing and a key to understanding properties of the dynamical system that generate the time series. Efficient encoding requires to extracts features of the data and decompose the given data with the features. Features on time series can be represented as a set of kernel functions, and a given data can be decomposed using the kernel functions and sparsification of the representation. This sparse and shiftable kernel method efficiently encodes one-dimensional time series data into a series of point on time as a point process. Reservoir computing paradigm provides a strategy to model multi-dimensional time series data with randomly coupled non-linear elements. Here we propose to combine the shiftable kernel method and the reservoir computing paradigm. We use the echo state network, one of implementation of the reservoir computing, and propose that the reservoir computing can be utilized to dynamically organize the kernel functions and to efficiently encode multi-dimensional time series. We demonstrate that a complex multi-dimensional time series data can be encoded into a few points in the point process with the proposed method.

1. Introduction

Dynamical system modeling has played a crucial role in the development of techniques for data processing of many research fields. For instance, nonstationary acoustic structures of timing relations among acoustic events or harmonic periodicities provide cues for many types of auditory processing, e.g. sound localization and speech recognition. Time series data reflecting activities of the brain or a local neural circuit provide information underlying neural processing and are applicable for brain-machine interface. These time series data reflect properties of underlying complex dynamical system and may include redundancy and exhibit repetition of a certain pattern of time course. Reducing the redundancy, extracting features, and encoding the data are key to understand the underlying dynamical systems and may provide a basis of applications for pattern recognition, regression, and prediction.

Temporal features on time series can be characterized to a precise point in time, such as the onset of sound or a task cue. Signal decomposition using a linear superposition of time-shiftable kernel function [1, 2] works well to extract features of the data and to encode the data into point process. This approach works well particularly on sound data recorded as a one-dimensional time series.

Most of the above-mentioned time series data are recorded with multiple sensors, e.g. electroencephalogram, multi-unit recording of spike trains, and imaging data containing many pixels, and microphone array. Even sound data recorded as one-dimensional time series are often analyzed as multi-dimensional time series of the spectrogram. Each channel of these multi-dimensional time series may have temporal and spatial correlations with each other reflecting the structure of an underlying dynamical system.

The key of encoding such multi-dimensional data is to reduce redundancy among the channels. In one view, efficient coding theory, the goal of the modeling is to encode the maximal amount of information about the input signal by statistically independent features. Current methods like principal component analysis or independent component analysis perform this processing with an assumed spatial distributions of the data. However, there is no efficient coding strategy based on both spatial and temporal feature of multi-dimensional time series data.

Reservoir computing is a paradigm of understanding and training recurrent neural networks[3], where neurons are sparsely and randomly connected. In the reservoir computing, supervised adaptation of all weight value of recurrent connection is not necessary, and only training a memoryless supervised readout from the dynamical reservoir (recurrent network) is enough to obtain flexible and multiple time courses. One form of the reservoir computing paradigm is the echo state network[4, 5], which composed of simple non-linear elements.

Here we propose that the reservoir computing paradigm can be utilized to encoding multi-dimensional time series and that kernel functions for extracting features of a given time series data can be generated with dynamics of the echo state network. In this paper, we explain the generative form of the model and algorithm for the encoding, then show preliminary results.

2. Methods

Here we use the echo state network [4, 5] and the sparse and shiftable kernel method of signal representation [1, 2]. The basic idea of our approach is to dynamically organize a set of kernel functions with the echo state network and to efficiently represent the correlated multi-dimensional time series data with this set of kernel functions.

2.1. Generative Model with Reservoir Computing

In the proposed model (Figure 1), a given N_y-dimensional signal $\mathbf{d}(n)$ is represented by the output of the dynamical reservoir $\mathbf{y}(n)$. State of the dynamical reservoir $\mathbf{x}(n)$ and output $\mathbf{y}(n)$ are updated according to following equations.

$$\mathbf{x}(n+1) = \mathbf{x}(n) + \frac{\delta}{\tau}\Big[-\alpha_0\mathbf{x}(n) + \mathbf{f}_x\Big(\mathbf{W}^{rec}\mathbf{x}(n) + \mathbf{W}^{back}\mathbf{y}(n) + \mathbf{W}^{in}\mathbf{u}(n)\Big)\Big], \quad (1)$$

$$\mathbf{y}(n) = \mathbf{f}_y(\mathbf{W}^{out}\mathbf{x}(n)), \quad (2)$$

where $\mathbf{W}^{rec}$ is $N_x \times N_x$ weight matrices of recurrent connections on the dynamical reservoir, $\mathbf{W}^{back}$ is $N_x \times N_y$ weight matrices of feedback connections from the output layer to the dynamical reservoir, $\mathbf{W}^{in}$ is $N_x \times N_u$ weight matrices of input connections on from the input layer to the dynamical reservoir. We define $f_x(x) = f_y(x) = \tanh x$. $\delta = 1$, τ specifies the time scale of the dynamical reservoir.

The dynamical reservoir is driven by the input $\mathbf{u}(n)$, which is given as an ensemble of a kernel function ϕ. Here we use Gaussian function as the kernel $\phi(n) = \exp(-n^2/\sigma^2)$ with the width of kernel σ, and k th component of $\mathbf{u}(n)$ is represented as follows.

$$u_k = \sum_i^{n_k} s_i^k \phi(n - \tau_i^k), \quad (3)$$

where τ_i^k and s_i^k are the temporal position and coefficient of the i th instance of k th component of $\mathbf{u}(n)$. n_k is the number of instance on the k th component of $\mathbf{u}(n)$.

2.2. Encoding Algorithms

Above described equations specify the generative form of the model but does not provide an encoding algorithm. The optimal values of τ_i^k, s_i^k, and $\mathbf{W}^{out}$ for a given signal are necessary to be computed with randomly generated $\mathbf{W}^{rec}$, $\mathbf{W}^{back}$, and $\mathbf{W}^{in}$.

As to the first step for generating $\mathbf{W}^{rec}$, $\mathbf{W}^{back}$, and $\mathbf{W}^{in}$, we follows the procedures in [5]. For $\mathbf{W}^{rec}$, first, randomly generate $N_x \times N_x$ matrix$\mathbf{W}_0$ by assigning 1 or -1 to randomly selected $\beta_r N_x \times N_x$ components of $\mathbf{W}_0$ and assign 0 to the others. Then, normalize $\mathbf{W}_0$ to a matrix W_1 with unit spectral radius by putting $\mathbf{W}_1 = 1/|\lambda_{max}|\mathbf{W}_0$, where

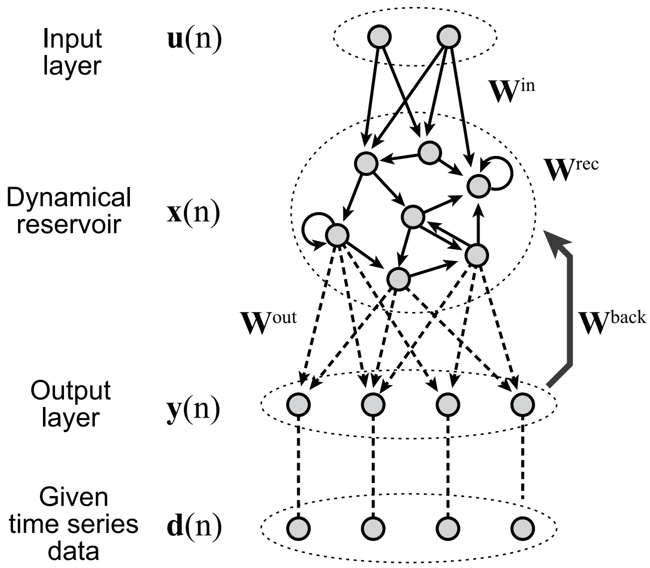

Figure 1: Structure of proposed model

λ_{max} is maximum eigenvalue of W_0. Finally, scale $\mathbf{W}_1$ to $\mathbf{W}^{rec} = \alpha_r\mathbf{W}_1$. For $\mathbf{W}^{back}$, first, randomly assign $-1 \sim 1$ to $\beta_b N_x \times N_y$ components of $\mathbf{W}_3$ with a uniform distribution and assign 0 to the other units. Then, normalize the sum of columns of the unit vectors, multiply with coefficient α_b, and set it to $\mathbf{W}^{back}$. $\mathbf{W}^{in}$ is generated by same manner with coefficient α_i and β_i

In the second step, optimal values of τ_i^k, s_i^k, and $\mathbf{W}^{out}$ are computed with a given time series data. The objective of this optimization is to minimize the error while maximizing coding efficiency. Here we use the following equation as the objective function.

$$E = \sum_i^{N_y}\sum_n (y_i(n) - d_i(n))^2 + \lambda\sum_k^{N_u}\sum_n |u_k(n)| \quad (4)$$

The first term on the r.h.s of equation is the error and the second term is a term for enhancing sparseness of the input.

We try to minimize E by finding optimal list of instances (τ_i^k and s_i^k) with iterations described below. In each iteration step, find the optimal τ_i^k and s_i^k with brute force approach, namely adopt $(\tau_i^k, s_i^k) = \arg\min_{\tau_i^k, s_i^k} E$ as an instance on the input layer.

In each iteration step, $\mathbf{W}^{out}$ are computed by driving the model with "teacher-focing", which means that the feedback of output state $\mathbf{y}$(n) is replaced with the teacher signal (the given input) $\mathbf{d}$(n) as following equations.

$$\mathbf{x}(n+1) = \mathbf{x}(n) + \frac{\delta}{\tau}\Big[-\alpha_0\mathbf{x}(n) + \mathbf{f}_x\Big(\mathbf{W}^{rec}\mathbf{x}(n) + \mathbf{W}^{back}\mathbf{d}(n) + \mathbf{W}^{in}\mathbf{u}(n)\Big)\Big]. \quad (5)$$

$\mathbf{x}(n)$ and $\mathbf{y}(n)$ in time from T_0 to T_1 are used to calculate

$\mathbf{W}^{out}$. Align the time series of $\mathbf{x}(n)$ into state collecting matrix$M$, where M is $(T_1 - T_0 - 1) \times N_x$ matrix. Then, input sigmoid-inverted $\mathbf{d}(n)$, or $\tanh^{-1} \mathbf{d}(n)$ into G, where G is $(T_1 - T_0 - 1) \times N_y$ matrix. $\mathbf{W}^{out}$ is calculated using pseudo-inverse of M as

$$(\mathbf{W}^{out})^T = M^{-1}G. \tag{6}$$

Continue the iteration till the E does not decrease, and the set of instances of the input layer is obtained.

3. Results

We demonstrate our methods with an artificially generated multi-dimensional time series (Figure 2). This time series is generated as an ensemble of spatially and temporally distributed Gaussian functions on $N_y = 20$ dimension of given data (See red curves in Fig. 2(c)).

We use the dynamical reservoir of size $N_x = 100$ and $N_u = 5$ input layer with following parameter values. $\alpha_r = 0.2, \beta_r = 0.1, \alpha_b = 0.2, \beta_b = 0.1, \alpha_i = 1, \beta_i = 0.1, \tau = 2.5, \delta = 1, \sigma = 5$, and $\lambda = 1.0$. For simplicity, we set $s_i^k = 1$ for this simulation.

Figure 2 shows that the given multi-dimensional time series is encoded with three instances on the input layer, which virtually corresponds to three points of marked point process (Fig. 2(a)). The input drives the dynamical reservoir and causes specific pattern of fluctuations (Fig. 2(b)). This fluctuation is read out to the output layer so that the output layer reproduces the given time series data (Fig. 2(c)). The responses on the output layer well reproducing the given time series data (Fig. 2(d)).

The proposed method uses random numbers for generating weight matrix in the first step of the encoding, and thus the performance of the encoding depend on the realization of the weight matrix, whereas, the performance is stable for this specific time series data, and result shown in Fig. 2 is a typical case.

4. Discussion

Here we propose a new encoding method for multi-dimensional time series data based on reservoir computing and sparse and shiftable kernel method. Our preliminary result shows that complex multi-dimensional time series can be virtually encoded into a few points in the point process.

The present study may provide insights for understanding efficient coding mechanisms of biological systems, i.e. auditory system, particularly, the higher order representation of auditory coding and animal vocalizations.

In the future, we should evaluate details of the proposed method, namely, the dependency of the performances on the many parameters and network structure. To apply this method to large-scale and realistic problem, more efficient ways of configuring connection matrix and encoding algorithms are necessary. Furthermore, we should apply this method to real engineering problems, e.g. encoding audio signals, communication, analysis of biomedical data including applications for brain-machine interfaces.

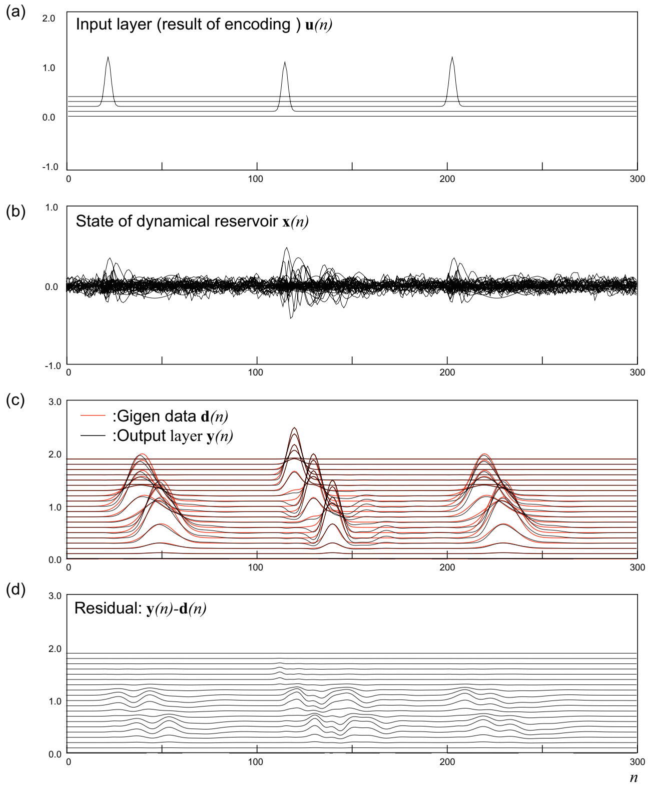

Figure 2: An example of encoding of multi-dimensional time series. (a) Input for the dynamical reservoir that is a result of encoding. (b) Responses in the dynamical reservoir. (c) Given time series data (red curves) and a responses of the output layer of the network (black curves). (d) Residual (error) of reproduced data. In (a), (c), and (d), multiple time series are displayed by shifting to position to the vertical direction.

Acknowledgments

The author would like to thank NOLTA2016 organizing committee members for their fruitful suggestions and comments. This work was supported by JSPS KAKENHI Grant Number 16K00246.

References

[1] M. S. Lewicki, "Efficient coding of natural sounds", *Nat. Neurosci.*, 5(4), pp. 356–363, 2002.

[2] M. S. Lewicki, "Efficient coding of time-varying patterns using spiking population code", In R. P. N. Rao, B. A. Olshausen, and M. S. Lewicki (Eds.), *Probabilis-*

tic models of the brain: perception and neural functions, pp.241–255, Cambridge, MA:MIT Press, 2002.

[3] M. Lukoševičius, H. Jaeger, B. Schrauwen, ”Reservoir Computing Trends”, *Knstliche Intelligenz*, 26(4), pp. 365-371, 2012.

[4] H. Jaeger and H. Haas, ”Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication” *Sci.*. vol.304, Issue 5667, pp.78–80, 2004.

[5] H. Jaeger, ”A tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the “echo state network ” approach”, *GMD Report*, German National Research Center for Information, pp.1-46, 2005.