International Symposium on Nonlinear Theory and its Applications
Xi'an, PRC, October 7–11, 2002

Prediction of Time Series Based on a Mapping Function

Yasuhiro HIRAI and Toshihiro SHIMIZU

Graduate School of Engineering
Kokushikan University
Setagaya 4-28-1 Setagaya-ku, Tokyo, Japan
Phone:+81-3-5481-3257, Fax:+81-3-5481-3253
Email: g382307b@kokushikan.ac.jp

1. Introduction

In our world, various types of time series exist. For example, they are a time series of stock price, pulse wave and earthquake wave, etc. We want to predict the future value of a system variable from past data in many situations of science and technology. To predict the future value, various methods have been proposed and their prediction ability has been discussed. Neural network models are often employed to predict the future value of time series, in which the error-back-propagation learning algorithm is used. In some neural network models, it has been proved that any continuous mapping can be approximately realized by the neural networks [1]. However the model, which make the long term prediction possible, has not yet been established. If there exists a mapping function which generates the time series and we can get the analytic form, we can easily predict the future value of the time series over considerably long time.

In this paper we propose a new method for prediction of complicated time series. The method is based on the assumption that there exists a mapping function which generates the complicated time series. In some cases we can expand the mapping function in Taylor's series. The coefficients of Taylor's expansion are determined by the gradient descent method.

It depends much on the dimension of the system how to assume the structure of the mapping function. Let us consider a system which is described by several variables. According to Takens' theorem [2] we can reconstruct the original attractor only from the information about one variable by using the delay time method. This fact may suggest that the future value x_{n+1} can be predicted by using the mapping function, which is a function of previous values $(x_n, x_{n-1}, \cdots x_{n-m+1})$, if the embedding dimension is given by m.

The method is tested for two simple chaotic time series which are generated by Logistic map and Hénon map. In these simple cases we can show that the expansion coefficients can be determined precisely. The method is also applied to a realistic time series.

We explain the method in section 2. The results of simulations are discussed in section 3. We give the conclusion in section 4.

2. Method

For simplicity we first consider a time series which is described by one variable: $\{x_n\}(n = 1, 2 \cdots N)$. Then we assume that there exists a mapping function, which generates x_{n+1} from x_n. We determine the function as follows. We expand the mapping function $f(y)$ in Taylor's series:

$$f(y) = \sum_{k=0}^{\infty} c_k y^k. \tag{1}$$

We define the cost function as follows:

$$E = \frac{1}{2}(x_{n+1} - f(x_n))^2 \qquad (n = 1, 2 \cdots N). \tag{2}$$

In this method the coefficients $\{c_k\}$ are so determined successively that the cost function E is minimized by using the gradient descent method. The coefficients are updated according to the rule:

$$c_k^{new} = c_k^{old} - \eta \frac{\partial E}{\partial c_k}, \tag{3}$$

where η is a positive small constant.

This method can be easily generalized to many variable case.

3. Simulation

We first apply the method to simple chaotic time series which are generated by Logistic map and Hénon map. Then we apply it to the realistic time series.

3.1. 1-D time series

We discuss a time series described by one variable. The efficiency of the method is tested for simple chaotic time series which is generated by the logistic map:

$$x_{n+1} = 4x_n(1 - x_n). \tag{4}$$

The mapping function which we want to obtain is defined as follows:

$$f(x) = \sum_{l=0}^{k} c_l x^l. \tag{5}$$

The initial values of coefficients $\{c_k\}$ are given randomly between -0.01 and 0.01. η is fixed as 0.10.

We first apply the method to the case of $k = 3$. Figure 1 shows the time evolution of cost function. The cost function E becomes zero after about 2×10^6 steps. Then we can obtain $c_0 = 0, c_1 = 4, c_2 = 4$ and $c_3 = 0$. Consequently the mapping function $f(x) = 4x - 4x^2$ is determined precisely.

Figure 1: Time evolution of cost function

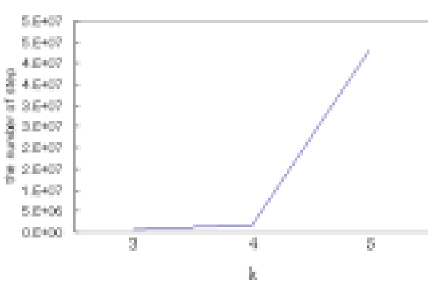

Figure 2: k - dependence of convergence

Next we study the k - dependence of convergence, changing k from 3 to 5. Figure 2 shows the necessary convergence step when k is changed from 3 to 5. In each case, the mapping function $f(x) = 4x - 4x^2$ is obtained precisely. The number of step is increased extremely when order k is increased. In the case of Logistic map k must be greater than 2 in eq.(5). However, we must retain higher order terms in more general cases. Then we need longer convergence step.

3.2. 2-D time series

We next consider a time series which consists of two variables. Such an example is given by Hénon map:

$$x_{n+1} = 1 - 1.4x_n^2 + y_n, \tag{6}$$
$$y_{n+1} = 0.3x_n. \tag{7}$$

In this case we introduce two mapping functions as follows:

$$f(x, y) = c_0x^0 + c_1x^1 + c_2x^2 + c_3x^3 + c_4y^1 + c_5y^2 + c_6y^3 + c_7x^1y^1 + c_8x^2y^1 + c_9x^1y^2, \tag{8}$$
$$g(x, y) = d_0x^0 + d_1x^1 + d_2x^2 + d_3x^3 + d_4y^1 + d_5y^2 + d_6y^3 + d_7x^1y^1 + d_8x^2y^1 + d_9x^1y^2. \tag{9}$$

The initial values of coefficients $\{c_k\}$ and $\{d_k\}$ are given randomly between -0.01 and 0.01. η is fixed as 0.01.

When the cost function E reaches to zero after about 1×10^8 steps, we obtain $c_0 = 1, c_2 = -1.4, c_4 = 1, d_1 = 0.3$ and other coefficients are zero. The mapping functions $f(x, y) = 1 - 1.4x^2 + y$ and $g(x, y) = 0.3x$ are obtained precisely.

From eqs.(6) and (7), x_{n+1} can be expressed in terms of x_n and x_{n-1} in Hénon map:

$$x_{n+1} = 1 - 1.4x_n^2 + 0.3x_{n-1}. \tag{10}$$

Therefore we can consider a mapping function of the form $f(x, y)$ with two variables $x = x_n$ and $y = x_{n-1}$.

$$f(x, y) = c_0x^0 + c_1x^1 + c_2x^2 + c_3x^3 + c_4y^1 + c_5y^2 + c_6y^3 + c_7x^1y^1 + c_8x^2y^1 + c_9x^1y^2. \tag{11}$$

The initial values of coefficients $\{c_k\}$ are given randomly between -0.01 and 0.01. η is fixed as 0.10.

The cost function E reaches to zero after about 1×10^5 steps. Then we obtain $c_0 = 1.0$, $c_2 = -1.4$, $c_4 = 0.3$. The mapping function $f(x_n, x_{n-1}) = 1.0 - 1.4x_n^2 + 0.3x_{n-1}$ is obtained precisely. In this case the convergence time is shorter than in previous case.

These results seem to show that our method is useful also for 2-D time series. As has been discussed in section 1, the mapping function may generally be a function of the previous values $(x_n, x_{n-1}, \cdots x_{n-m+1})$. This assertion was shown to be valid for Hénon map with $m = 2$.

3.3. Mapping function with arbitrary embedding dimension

In the previous section we have discussed only the case of the embedding dimension $m = 1$ (Logistic map) and $m = 2$ (Hénon map). In general cases, however, we have a priori no information about the embedding dimension. In this section we study whether or not we can get the correct coefficients for other embedding dimension m except for $m = 1$ (Logistic map) and $m = 2$ (Hénon map).

First we test the method for the time series, which is generated by Logistic map, by starting with $m = 5$. The mapping function is given by

$$x_{n+1} = f(x_n, x_{n-1}, x_{n-2}, x_{n-3}, x_{n-4}). \tag{12}$$

The mapping function has 56 coefficients if the terms up to third order are retained in Taylor's expansion of eq.(12). The initial values of coefficients $\{c_k\}$ are given randomly between -0.001 and 0.001. η is fixed as 0.10.

We have repeated 5×10^7 iteration as the learning process, and the cost function has become lower than 0.001. We have at first expected that the coefficients of the terms x_n and x_n^2 are 4 and -4, respectively and other coefficients are zero after the learning process. According to the numerical results, we have found that many coefficients are different from zero. This means that the obtained mapping function is different from eq.(4). As shown in figure 3, however, the obtained mapping function can predict the time series over about 20 steps.

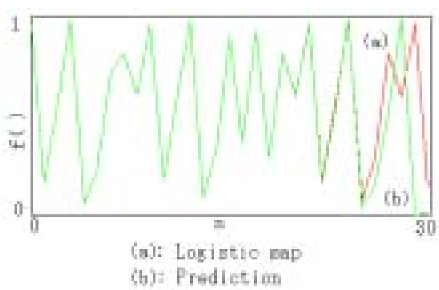

Figure 3: Prediction in Logistic map

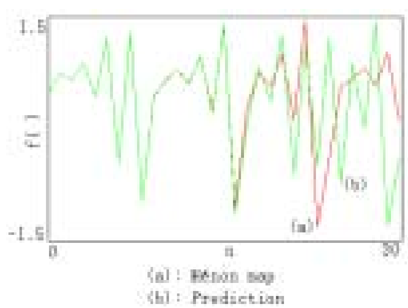

Figure 4: Prediction in Hénon map

Next, we test the method for the time series that is generated by Hénon map of eq.(10). We use eq.(12) as the mapping function. The initial values of coefficients are given randomly between -0.0001 and 0.0001. η is fixed as 0.01.

We have got the mapping function, which is different from eq.(10) also in this case. Figure 4 shows the prediction after 2×10^6 steps. In this case we can predict the time series over about 15 steps.

3.4. Prediction with small data

So far we have tried to determine the coefficients of the mapping function by using many data, whose number is equivalent to that of the learning steps. In actual systems, however, we often encounter many cases, in which we can get only limited number of data, i.e., $100 \sim 500$ data. In this section we study whether or not the mapping function can be determined by using only the limited number of data, i.e., 100 data in Logistic map and in Hénon map. As the mapping function we have assumed eq.(12).

In the case of Logistic map the initial values of coefficients are given randomly between -0.001 and 0.001. η is fixed as 0.10.

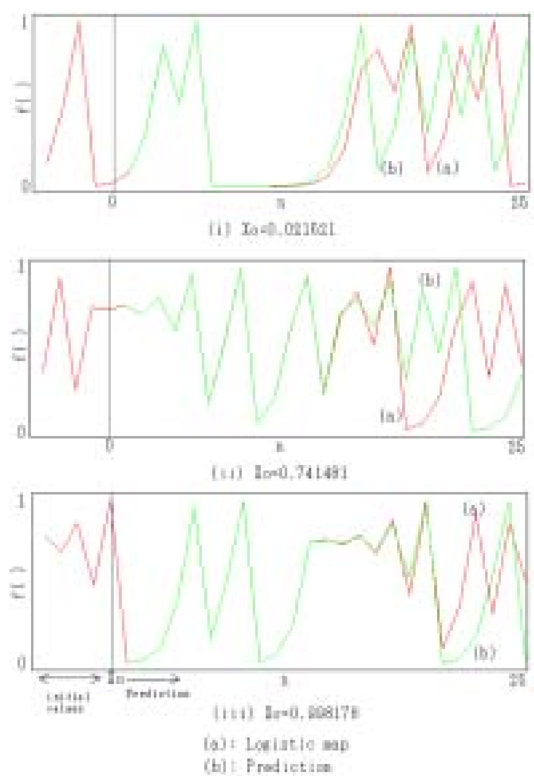

Figure 5: Prediction in Logistic map

Figure 5 shows the prediction after 3×10^5 iterations as the learning process by using 100 different inputs for three different initial values x_0. As shown in figure 5, the obtained mapping function can predict the time series over about 10 steps for different initial values.

In the case of Hénon map the initial values of coefficients are given randomly between -0.0001 and 0.0001. η is fixed as 0.01.

Figure 6 shows the prediction in Hénon map after 5×10^5 iterations as the learning process by using 100 different inputs for three different initial values x_0. As shown in figure 6, the obtained mapping function can predict the time series over about 20 steps.

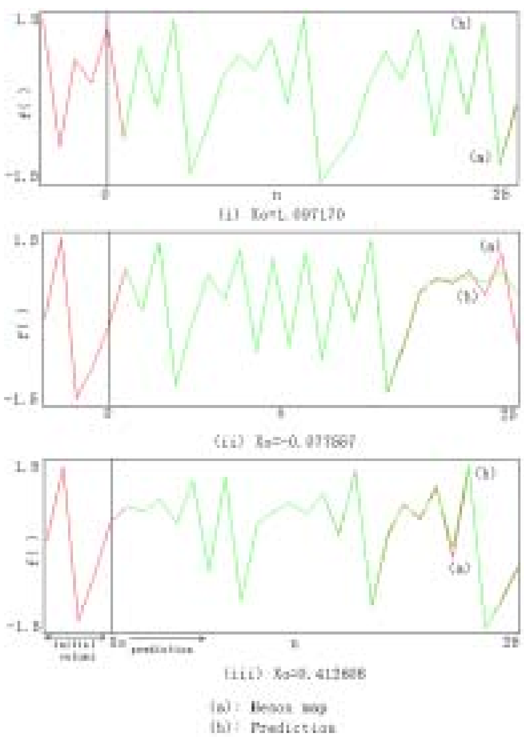

Figure 6: Prediction in Hénon map

These results show that the mapping function can be determined satisfactorily by using small data.

3.5. Realistic time series

We apply the method to the realistic time series which is the exchange rate between US dollar and Japanese yen. This time series has 348 data. The each data of time series is normalized between -1 and 1. We introduce eq.(12) as the mapping function. The initial values of coefficients are given randomly between -0.001 and 0.001. η is fixed as 0.01.

Figure 7 shows the output of the mapping function obtained after 1×10^7 iterations. We give the initial values of the realistic time series to the obtaind mapping function. We compare the output of the mapping function

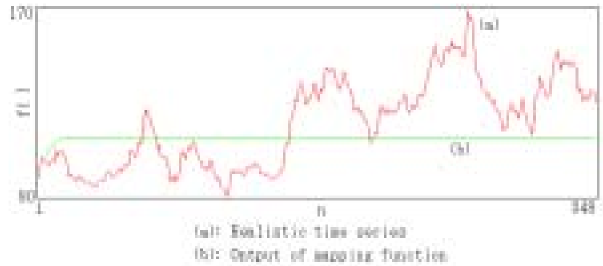

Figure 7: Output of the mapping function in realistic time series

with the realistic time series. It means that the mapping function cannot learn the time series even after 1×10^7 iterations. This suggests that the embedding dimension of the actual economic time series is considerably high and we must take higher embedding dimension m than $m = 5$.

4. Conclusion

We have proposed a new method which is based on the assumption that there exists a mapping function which generates the complicated time series. The method was applied to two types of simple chaotic time series. As a result, the coefficients of the mapping function could be determined precisely; the coefficients of the mapping function became the same values as the bifurcation parameters in each case. In the case of high dimensional time series the mapping function with the previous values $(x_n, x_{n-1}, \cdots x_{n-m+1})$ was shown to be effective in Hénon map.

In the realistic time series we could not obtain the mapping function with $m = 5$ which can generate the time series. The reason may come from the fact that the embedding dimension of the actual economic time series is considerably high. If we take the mapping function with higher m, however, we can get the effective result.

References

[1] K. Funahashi, "On the Approximate Realization of Continuous Mappings by Neural Networks," Neural Networks, Vol. 2, pp. 183-192, 1987.

[2] F. Takens, "Detecting strange attractors in turbulence," in Lecture Notes in Mathematics, No. 898, pp. 366-381, 1981.