Bootstrap nonlinear prediction

Daisuke Haraki,[1] Tomoya Suzuki,[2] Hiroki Hashiguchi,[1] and Tohru Ikeguchi[1]
[1]*Graduate School of Science and Engineering, Saitama University, 225 Shimo-Ohkubo, Sakura-ku, Saitama-city 338-8570, Japan*
[2]*Department of Information Systems Design, Doshisya University, 1-3 Tatara Miyakodani, Kyotanabe-city 610-0394, Japan*
(Received 24 December 2005; revised manuscript received 15 January 2007; published 22 May 2007)

Estimating the Jacobian matrix of a nonlinear dynamical system through observed time-series data is one of the important steps in predicting future states of the time series. The Jacobian matrix is estimated using local information about divergences of nearby trajectories. Although the basic algorithm for estimating the Jacobian matrix generally works well, it often fails for short or noisy data series. In this paper, we proposed a scheme to effectively use near-neighbor information for more accurate estimation of the Jacobian matrix using the bootstrap resampling method. Then, to confirm the validity of the proposed method, we applied it to a mathematical model and several real time series. As a result, we confirmed that the proposed method greatly improves nonlinear predictability, not only for noise-corrupted mathematical models but also for real time series.

DOI: 10.1103/PhysRevE.75.056212 PACS number(s): 05.45.Tp, 02.70.−c, 05.10.−a

I. INTRODUCTION

To predict complex time series, several methods have been proposed, such as the Lorenz method of analogs [1], the Jacobian-matrix estimate prediction [2–5], triangulation [6], radial basis functions [7,8], and so on. If we use these prediction methods efficiently, it might be possible to predict real time series, such as vowel signals [9], the number of infectious disease patients [6], pulse waves [10], and other complex phenomena, more accurately. Then, it is inevitable to involve nonlinear prediction, even though its algorithms become more complicated, because such real time series are often generated by nonlinear dynamical systems.

Most nonlinear predictions are classified into global [7,8] or local methods [1–4,6,11]. In this paper, we focus on the Jacobian estimate prediction [2,3,5], which is categorized in a local linear prediction method. In such methods, it is important to utilize the local information of a predicted point as efficiently as possible. If an observed time series is short or is disturbed by noise, the reliability of local information becomes poor. In these cases, the estimated Jacobian matrices are unreliable, and it is almost impossible to achieve higher predictability.

To solve the issue, we applied a bootstrap method [12] to effectively use near-neighbor trajectories for constructing local linear predictors. In statistics, there are several studies for evaluating model parameters by the bootstrap method [13,14]. Unlike these approaches, we simply apply a bootstrap resampling scheme to select good near-neighbor information when we predict nonlinear, complex, possibly chaotic behavior. This method is useful for statistically producing a wide variety of near neighbors as new local information for precisely approximating Jacobian matrices in order to realize higher nonlinear prediction accuracy.

To confirm the validity of the proposed method, first we applied our method to the Ikeda map [15] as a mathematical model. The results show that the proposed method exhibits better predictability than conventional algorithms, and the proposed method is also effective in the cases of predicting short or noisy time series. Next, we applied the proposed method to analyze the nonlinearity of several real time-series data including the Japanese vowel /a/ [9], the number of measles patients [6], and the number of chickenpox patients [6]. As a result, we find that the proposed method is more effective for prediction as the prediction step gets larger.

This paper is organized as follows. In the next section, the conventional local linear prediction methods are reviewed. Section III introduces several problems of conventional prediction approaches, and we propose a different prediction method based on the bootstrap method [12]. Section IV shows the simulation results of applying the proposed method to several time-series data, and then Sec. V discusses the validity of the proposed method using the results of the simulations. Section VI concludes this paper.

II. LOCAL LINEAR PREDICTION METHODS

A. The Jacobian-matrix estimate prediction

1. Estimation of the Jacobian matrix

Let us consider a nonlinear dynamical system

$$\boldsymbol{x}(t+1) = \boldsymbol{F}(\boldsymbol{x}(t)), \tag{1}$$

where $\boldsymbol{F}$ is a d-dimensional nonlinear map and $\boldsymbol{x}(t)$ is a d-dimensional state at time t. To estimate the Jacobian matrix of $\boldsymbol{F}$ [2,5], we linearize Eq. (1) as follows:

$$\delta\boldsymbol{x}(t+1) = \boldsymbol{J}(\boldsymbol{x}(t))\,\delta\boldsymbol{x}(t), \tag{2}$$

where $\boldsymbol{J}(\boldsymbol{x}(t))$ is the Jacobian matrix at $\boldsymbol{x}(t)$, and $\delta\boldsymbol{x}(t)$ is a very small displacement vector from $\boldsymbol{x}(t)$. To evaluate $\boldsymbol{J}(\boldsymbol{x}(t))$ with only information about $\boldsymbol{x}(t)$, first, we extract a near-neighbor set of $\boldsymbol{x}(t)$. Let us denote the ith near neighbor of $\boldsymbol{x}(t)$ as $\boldsymbol{x}(t_{k_i})$, where $i=1,2,\ldots,M$. Here, M is the total number of near neighbors. After a temporal evolution, $\boldsymbol{x}(t)$ and $\boldsymbol{x}(t_{k_i})$ evolve into $\boldsymbol{x}(t+1)$ and $\boldsymbol{x}(t_{k_i}+1)$, respectively. Then, we denote the displacement vectors by $\boldsymbol{y}_i=\boldsymbol{x}(t_{k_i})-\boldsymbol{x}(t)$ and $\boldsymbol{z}_i=\boldsymbol{x}(t_{k_i}+1)-\boldsymbol{x}(t+1)$. Here, $\boldsymbol{y}_i$ corresponds to $\delta\boldsymbol{x}(t)$, and $\boldsymbol{z}_i$ corresponds to $\delta\boldsymbol{x}(t+1)$ in Eq. (2). If the norms of $\boldsymbol{y}_i$ and $\boldsymbol{z}_i$ and the temporal evolution are small enough, we can approximate the relation between $\boldsymbol{z}_i$ and $\boldsymbol{y}_i$ with the following linear equation:

$$\boldsymbol{z}_i = \boldsymbol{G}(t)\boldsymbol{y}_i,$$

where the matrix $\boldsymbol{G}(t)$ corresponds to the Jacobian matrix $\boldsymbol{J}(\boldsymbol{x}(t))$ in Eq. (2). Then we estimate $\boldsymbol{G}(t)$ using least-square-error fitting, which minimizes the average square error S:

$$S = \frac{1}{M}\sum_{i=1}^{M} |\boldsymbol{z}_i - \boldsymbol{G}(t)\boldsymbol{y}_i|^2.$$

In other words, we can estimate $\boldsymbol{G}(t)$ using the following equations:

$$\boldsymbol{G}(t)\boldsymbol{W} = \boldsymbol{C}, \tag{3}$$

where $\boldsymbol{W}$ is the variance matrix of $\boldsymbol{y}_i$, and $\boldsymbol{C}$ is the covariance matrix between $\boldsymbol{y}_i$ and $\boldsymbol{z}_i$. If $\boldsymbol{W}$ has an inverse matrix, we can obtain $\boldsymbol{G}(t)$ from $\boldsymbol{G}(t)=\boldsymbol{C}\boldsymbol{W}^{-1}$.

2. Prediction based on the Jacobian-matrix estimation

Let us introduce the process of a nonlinear prediction of $\boldsymbol{x}(t)$ on an attractor from the dynamical system of Eq. (1) [3]. Our purpose is to predict the s-step future $\boldsymbol{x}(t+s)$ of $\boldsymbol{x}(t)$. Because we do not know a future value $\boldsymbol{x}(t)$, we cannot use the information about $\boldsymbol{z}_i$, and thus cannot have direct information about $\boldsymbol{G}(t)$. To solve the problem, we use the information of the nearest neighbor $\boldsymbol{x}(t_{k_0})$ of $\boldsymbol{x}(t)$. Then we calculate a displacement vector

$$\boldsymbol{y}' = \boldsymbol{x}(t) - \boldsymbol{x}(t_{k_0}).$$

Next, we can estimate the Jacobian matrix $\boldsymbol{G}(t_{k_0})$ at $\boldsymbol{x}(t_{k_0})$ by following the process introduced in Sec. II A 1. If we define $\hat{\boldsymbol{x}}(t+1)$ as the predicted future value of $\boldsymbol{x}(t)$, we can denote the predicted displacement vector $\hat{\boldsymbol{z}}'=\hat{\boldsymbol{x}}(t+1)-\boldsymbol{x}(t_{k_0}+1)$ by

$$\hat{\boldsymbol{z}}' = \boldsymbol{G}(t_{k_0})\boldsymbol{y}'.$$

Then we can predict $\hat{\boldsymbol{x}}(t+1)$ as follows:

$$\hat{\boldsymbol{x}}(t+1) = \boldsymbol{G}(t_{k_0})[\boldsymbol{x}(t) - \boldsymbol{x}(t_{k_0})] + \boldsymbol{x}(t_{k_0}+1). \tag{4}$$

The above prediction algorithm first estimates a Jacobian matrix at each point and then predicts a future value by approximating the temporal evolution of the local displacement vector. In other words, the algorithm uses the first-order local information of the first term in Eq. (4) as well as the zeroth-order local information of the second term in Eq. (4). Then, repeating the above scheme s times iteratively, we can predict the s-step future of $\boldsymbol{x}(t)$.

B. The Lorenz method of analogs and its modified prediction methods

In this paper, to check the validity of the proposed method, we compare its performance with other local linear prediction methods, including the Lorenz method of analogs [1]. First, in this method, we search for the nearest neighbor $\boldsymbol{x}(t_{k_0})$ of $\boldsymbol{x}(t)$ on the reconstructed attractor. Then we predict the s-step future of $\boldsymbol{x}(t)$, $\hat{\boldsymbol{x}}(t+s)$, as $\boldsymbol{x}(t_{k_0}+s)$. This method is very useful if we treat noiseless data. However, if the data series length is short or the data are disturbed by noise, this method does not always work well. To improve the problem, several modified concepts of the Lorenz method of analogs have been proposed. These modified concepts use near neighbors $\boldsymbol{x}(t_{k_i})$ $(i=0,1,\ldots,M)$, and make predictions by calculating the average of temporally evolved neighbors $\boldsymbol{x}(t_{k_i}+s)$ with different weighting strategies. For example, in Ref. [16], the future value of $\boldsymbol{x}(t)$ is predicted by

$$\hat{\boldsymbol{x}}(t+1) = \frac{\sum_{i=0}^{M} d_i^{-1}\boldsymbol{x}(t_{k_i}+1)}{\sum_{i=0}^{M} d_i^{-1}}, \tag{5}$$

where $d_i=|\boldsymbol{x}(t_{k_i})-\boldsymbol{x}(t)|$. On the other hand, Ref. [6] introduces a weighted average using negative exponents of d_i, and Eq. (5) is rewritten as

$$\hat{\boldsymbol{x}}(t+1) = \frac{\sum_{i=0}^{M} \exp(-d_i)\boldsymbol{x}(t_{k_i}+1)}{\sum_{i=0}^{M} \exp(-d_i)}.$$

III. BOOTSTRAP RESAMPLING METHOD FOR ESTIMATING RELIABLE JACOBIAN MATRICES

In the conventional Jacobian-matrix estimate prediction [2,3], if the number of near neighbors of $\boldsymbol{x}(t)$ is small, it is difficult to estimate reliable Jacobian matrices. In an actual time-series analysis, it is often the case that the number of an observed time series is small. Such undesirable situations would lead to unreliable estimation. To solve the issue, we introduced the bootstrap resampling scheme [12] to perform stable estimation of the Jacobian matrices by increasing the variety of near-neighbor sets.

The bootstrap method is useful for estimating the characteristics of a population from a small amount of observed data efficiently. First, we selected near neighbors of $\boldsymbol{x}(t)$ in the same way as the conventional method. The nearest neighbor of $\boldsymbol{x}(t)$ is denoted by $\boldsymbol{x}(t_{k_0})$, and the other neighbors are denoted by $\boldsymbol{D}_T=\{\boldsymbol{x}(t_{k_1}),\boldsymbol{x}(t_{k_2}),\ldots,\boldsymbol{x}(t_{k_M})\}$. We performed a sampling with the replacement of $\boldsymbol{D}_T$ to obtain a new set of near neighbors $\boldsymbol{D}_1^*=\{\boldsymbol{x}_1^*(t_{k_1}),\boldsymbol{x}_1^*(t_{k_2}),\ldots,\boldsymbol{x}_1^*(t_{k_M})\}$. Then we estimated the Jacobian matrix $\boldsymbol{G}_1^*(t_{k_0})$ at $\boldsymbol{x}(t_{k_0})$ using $\boldsymbol{D}_1^*$ as introduced in Sec. II A 1. Finally, we predicted a future point of $\boldsymbol{x}(t)$ by

$$\hat{\boldsymbol{x}}_1^*(t+1) = \boldsymbol{G}_1^*(t_{k_0})[\boldsymbol{x}(t) - \boldsymbol{x}(t_{k_0})] + \boldsymbol{x}(t_{k_0}+1).$$

We repeated such bootstrap estimates B times; namely, the bth bootstrap-predicted point is described by

$$\hat{\boldsymbol{x}}_b^*(t+1) = \boldsymbol{G}_b^*(t_{k_0})[\boldsymbol{x}(t) - \boldsymbol{x}(t_{k_0})] + \boldsymbol{x}(t_{k_0}+1),$$

where $b=1,2,\ldots,B$. Then we decided the final predicted point of $\boldsymbol{x}(t)$ by calculating its mean value as follows:

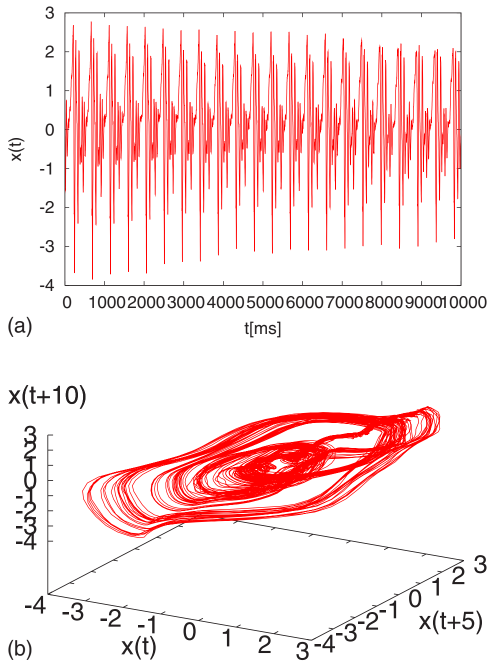

FIG. 1. (Color online) (a) Time-series data of the Japanese vowel /a/. The original sampling rate is 48 kHz. The mean value was subtracted from the original time series. (b) Reconstructed attractor of the Japanese vowel /a/ with $d=3$ and $\tau=5$.

$$\hat{\boldsymbol{x}}(t+1) = \frac{1}{B}\sum_{b=1}^{B} \hat{\boldsymbol{x}}_b^*(t+1).$$

If we estimate the Jacobian matrices by the resampling method described above, it is important not to apply the above algorithm blindly, because it is possible that the matrix $\boldsymbol{W}$ in Eq. (3) does not always have an inverse. This case exists because resampled near neighbors (the bootstrap samples) can be duplicated or clustered, making the matrix $\boldsymbol{W}$ not a full rank matrix.

This is one of the most important points we should consider in the present case, because application of the bootstrap resampling scheme to select good near neighbors and to improve the local linear predictability has different aspects from the conventional bootstrap method of estimating statistics. Even if we encounter such a case, we neither toss out the resampled near neighbors nor perform a new resampling, because the resampled near-neighbor sets with such a trick would be biased and break the condition that the members of the resampled set are independently and identically distributed.

To avoid such an undesirable situation in the present case, we performed the following procedure.

(1) Applying a diagonalization algorithm to $\boldsymbol{W}$, we calculated an orthogonal matrix $\boldsymbol{P}$ and a diagonal matrix $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_i)$ $(i=1,\ldots,d)$, where d is the dimension of $\boldsymbol{W}$ and λ_i is the ith eigenvalue of $\boldsymbol{W}$.

(2) The matrices of $\boldsymbol{W}$, $\boldsymbol{P}$, and $\boldsymbol{\Lambda}$ satisfy the relation $\boldsymbol{W} = \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^{-1}$.

(3) If $\lambda_i > 10^{-6}$ for all i, we calculated the inverse matrix by $\boldsymbol{W}^{-1} = \boldsymbol{P}\boldsymbol{\Lambda}^{-1}\boldsymbol{P}^T$, where $\boldsymbol{\Lambda}^{-1} = \mathrm{diag}(1/\lambda_i)$.

(4) If there exists an index i such that $\lambda_i < 10^{-6}$, which corresponds to the case that $\boldsymbol{W}$ is not of full rank, we calculated the inverse matrix by

$$\boldsymbol{W}^{-1} = \boldsymbol{P}\left(\begin{array}{ccc|c} 1/\lambda_1 & & & \\ & \ddots & & \\ & & 1/\lambda_{d-k} & \\ \hline & & & \boldsymbol{O} \end{array}\right)\boldsymbol{P}^T,$$

where k is the number of eigenvalues smaller than 10^{-6}. In this case, the bootstrap predicted point exists in the $(d\text{-}k)$-dimensional subspace.

In Sec. IV, we show that the above procedure works well to calculate the bootstrap-predicted points.

IV. SIMULATION RESULTS OF THE BOOTSTRAP PREDICTION METHOD

A. Data for simulations and criteria for estimating prediction accuracy

To confirm the validity of the proposed prediction method, we prepared several time-series data. First, we use the Ikeda map [15] as a mathematical model; it is described as follows:

$$x(t+1) = p + b\{x(t)\cos[\theta(t)] - y(t)\sin[\theta(t)]\},$$

$$y(t+1) = b\{x(t)\sin[\theta(t)] + y(t)\cos[\theta(t)]\},$$

$$\theta(t) = \kappa - \alpha/[1 + x^2(t) + y^2(t)],$$

where p, b, κ, and α are parameters. Generally, it is not easy to evaluate how the method can be applied to an observed time series, because it is impossible to obtain explicit information on the nonlinearity of the time series. The Ikeda map is suitable for checking the validity of the proposed method because it has higher-order nonlinearity even though it is a two-dimensional dynamical system. In the simulation, the parameters were set as $p=1.0$, $b=0.9$, $\kappa=0.4$, and $\alpha=6.0$. Then, we disturbed both $x(t)$ and $y(t)$ with Gaussian observational noise.

Moreover, we applied our method to several real time-series data. The first is the Japanese vowel /a/ (Fig. 1). This time series is suitable for benchmark tests because it has been analyzed and discussed in several studies [9,18–20]. Then, we applied our method to predict the number of measles [6] and chickenpox patients [6] (Fig. 2). Because these real time-series data are single-variable time series, we embedded the time series in a d-dimensional state space with a temporal delay τ using Takens' method [17] to perform nonlinear prediction. For estimating prediction error, we used the normalized root mean square error [21]

$$E = \frac{\sqrt{\langle[z(t) - \hat{z}(t)]^2\rangle}}{\sigma_z}, \tag{6}$$

where σ_z is the standard deviation of the time series $z(t)$, and $\hat{z}(t)$ is the predicted time series.

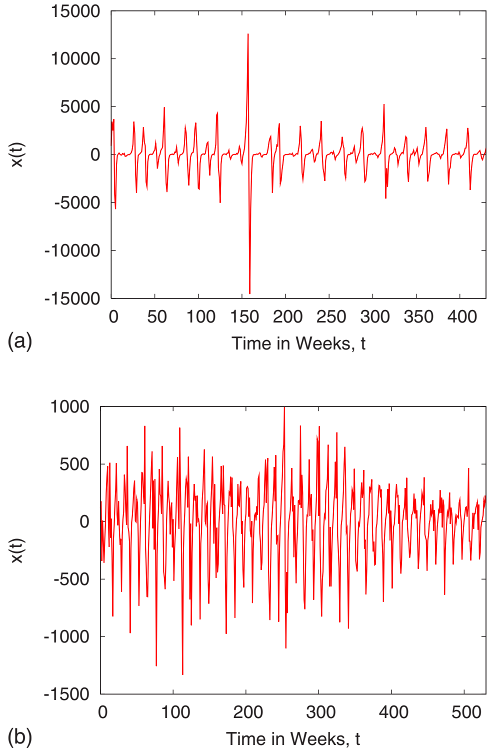

FIG. 2. (Color online) Time-series data of the difference of the number of (a) measles and (b) chickenpox patients [6].

Another measure for the improvement of the proposed method, which combines the Jacobian-matrix estimate prediction and the bootstrap method, is defined by

$$R = \frac{E_c - E_p}{E_c}, \tag{7}$$

where E_c means the root mean square error of the conventional methods and E_p means that of the proposed method. The measure estimates the ratio of the improvement between E_c and E_p. If R becomes a positive value, it means that the proposed method improves the prediction accuracy.

In this paper, to evaluate E_c and E_p, first we predicted the last half of time series of $z(t)$ using the first half of $z(t)$ as learning data [22]. Then we calculated the first improved ratio, which is described as R_1, with Eq. (7). Next, by changing the prediction and the learning parts, we predicted the first half of $z(t)$ using the learning data of the last half of $z(t)$. Then we calculated the second improved ratio R_2 with Eq. (7). We then use the mean value of R_1 and R_2 as the final R, which has statistical reliability.

B. Simulation results

1. Comparison of the proposed and several conventional prediction methods

In Fig. 3, we show the improved ratio R in the case of the Ikeda map [15]. Figures 4–6 show the results of the Ikeda map. We compared the proposed Jacobian-matrix estimate prediction based on the bootstrap resampling method with four conventional methods introduced in Sec. II: the original Jacobian-matrix estimate prediction, the Lorenz method of analogs [1] and its modification [16], and a weighted average using negative exponents [6]. In Fig. 3, the horizontal axis shows the number of time-series data N, and the vertical axis shows the improvement ratio R defined in Eq. (7). For simulations, we changed the neighborhood size of $\boldsymbol{x}(t_{k_0})$ whose radius is parametrized by the parameter r. Because the parameter r means the ratio of the radius of the neighbor size to the global attractor size, the total number of near neighbors M is different at each point on the attractors. The parameter of the resampling time B is 200. We use a different dimension d for each time series. In the case of predicting the Ikeda map, we use both $x(t)$ and $y(t)$ to construct a state space. In the case of predicting several real time series as mentioned below, we show the embedding dimensions d in the captions of Figs. 7–9 below.

Figure 3 shows the cases of $r=1\%$, 3%, 5%, and 10%, where the prediction step s is 1. These results show that each R always becomes positive; namely, it is very effective to apply the bootstrap resampling for the Jacobian-maxtrix estimate prediction. Although the improvement ratios are lower for the original Jacobian-matrix estimate prediction than the other conventional methods, such as the Lorenz method of analogs, the weighted average prediction, or the negative exponent prediction, the original Jacobian-matrix estimate prediction originally has higher prediction performance than the other conventional prediction methods. In the next section, by changing the prediction step s, we examine the performance of the proposed method for long-term predictability in more detail.

2. Performance of the proposed method in the case of large prediction steps

We compared the prediction accuracy of the original Jacobian-matrix estimate prediction and the proposed Jacobian-matrix estimate prediction based on the bootstrap resampling method from the viewpoint of long-term predictability.

In Figs. 4–6, the horizontal plane shows the near-neighbor radius r and the prediction step s, and the vertical axis shows the improved ratio R of Eq. (7). In Fig. 4, we applied the proposed method to the Ikeda map whose data length is $N=100$, 200, 500, and 1000. Moreover, in Figs. 5 and 6, we applied the proposed method to noisy Ikeda maps disturbed by Gaussian observational noise. In this paper, the noise level is quantified by the signal-to-noise ratio (SNR), which is calculated by

$$R_{\mathrm{sn}} = 10 \log_{10}\frac{\sigma_d^2}{\sigma_n^2}$$

where σ_d^2 is the variance of the original data and σ_n^2 is the variance of the Gaussian noise.

Next, we applied the proposed method to predict several real time series: the Japanese vowel /a/ [9], and the difference of the number of measles [6] and chickenpox patients [6]. These results are shown in Figs. 7–9.

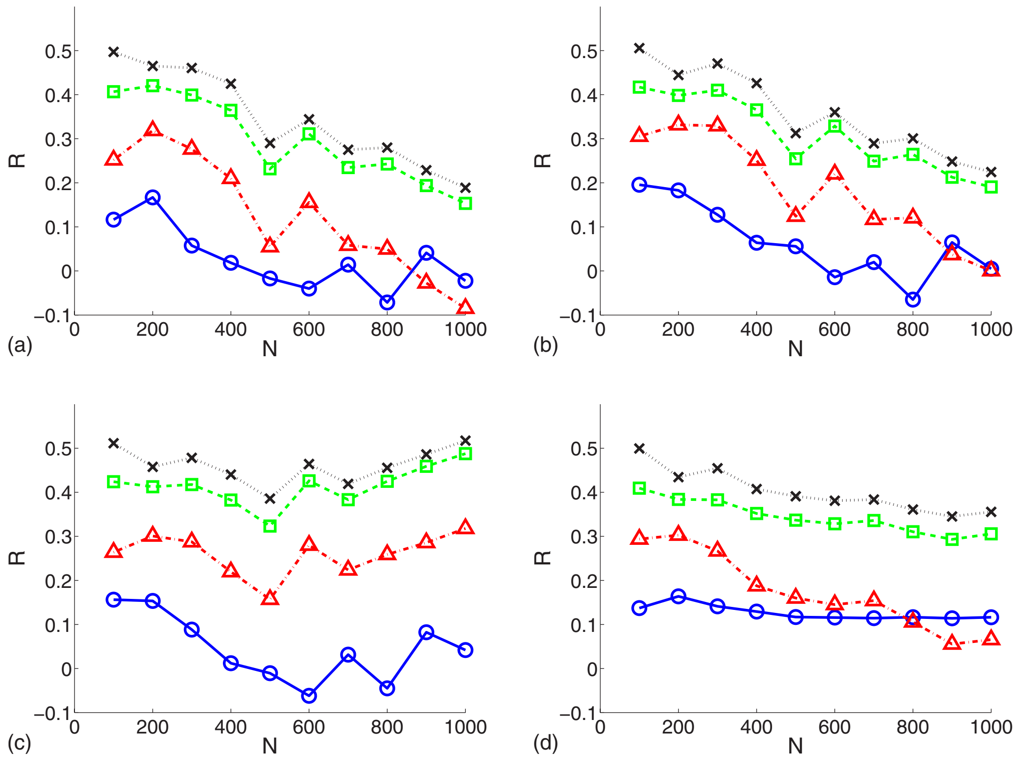

FIG. 3. (Color online) Improvement ratio R of the proposed prediction based on the bootstrap resampling prediction to several conventional prediction methods: the Jacobian-matrix estimate prediction (solid lines with circles), the Lorenz method of analogs (dotted lines with crosses), the weighted average prediction (dash-dotted lines with triangles), and the negative exponent prediction (dashed lines with rectangles). Predicted data are $(x(t),y(t))$ of the Ikeda map. The near-neighbor radius is set to (a) 1%, (b) 3%, (c) 5%, and (d) 10%, and the prediction step s is 1.

V. DISCUSSION

In Figs. 4–6, these results show that each R is almost positive, especially in the region of small r, that is, the bootstrap resampling method works well. On the other hand, in the region of large r, the improvement ratio R almost equals 0, that is, the root mean square error of the proposed method, E_p, is almost the same as that of the conventional method, E_c. The reason is that the estimation accuracy of the Jacobian matrices has a low prediction accuracy if the near-neighbor

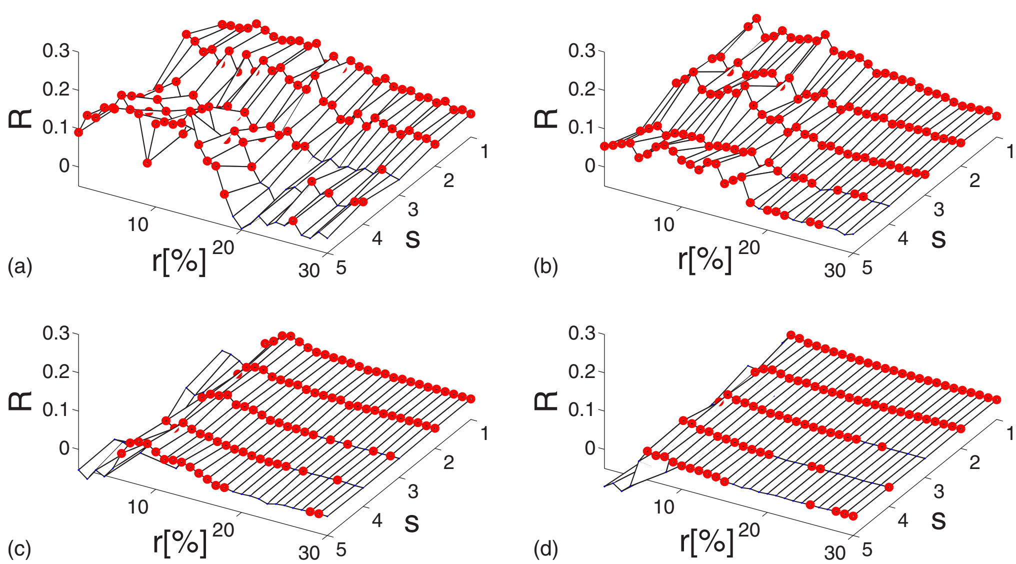

FIG. 4. (Color online) Properties of the improvement ratio R to the variables r and s. Improvements by the proposed method are indicated by closed circles. As the conventional method, we use the original Jacobian-matrix estimate prediction to the Ikeda map. N is the data length. N= (a) 100, (b) 200, (c) 500, and (d) 1000.

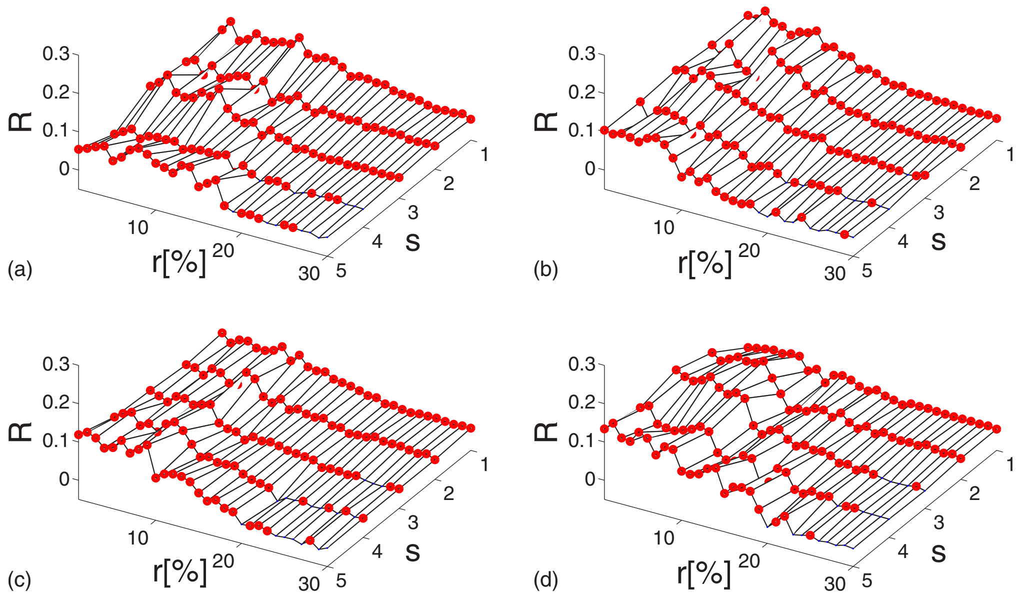

FIG. 5. (Color online) Properties of the improvement ratio R to the variables of r and s for an Ikeda map that is corrupted by observational noise with different noise levels: (a) noiseless, and R_{sn}= (b) 40, (c) 30, and (d) 20 dB. Improvements by the proposed method are indicated by closed circles. The data length N is fixed at 200.

radius is large, because in such a case the algorithm is not a local linear prediction. Then, application of the bootstrap method does not lead to improvement of the accuracy.

Moreover, from Figs. 5 and 6, we can confirm that the proposed method works well even if the time series is disturbed by large amounts of observational noise. Thus, the proposed bootstrap resampling method is effective in the cases where we cannot make r large, such as cases when we can observe just short time-series data or only noiseless time series. These cases often occur in real time-series analysis. Because we can confirm the validity of the proposed method using a mathematical model, we applied the method to predict real time-series data.

Figure 7 shows that the proposed method works better in the region of small r in the case of the Ikeda map. Moreover, these results show that the improvement ratio R depends on the embedding dimension d, that is, it is important to select an optimum embedding dimension. Then we applied our scheme to the difference of the number of measles and chickenpox patient. The results are shown in Figs. 8 and 9. They show almost the same tendencies as Fig. 7.

From these results, we can confirm that the proposed method is effective to predict real data as well. However, we also found that there exists a case that the improvement ratio R is small as shown in Fig. 5(b). By researching such cases in more detail, we could improve the proposed method.

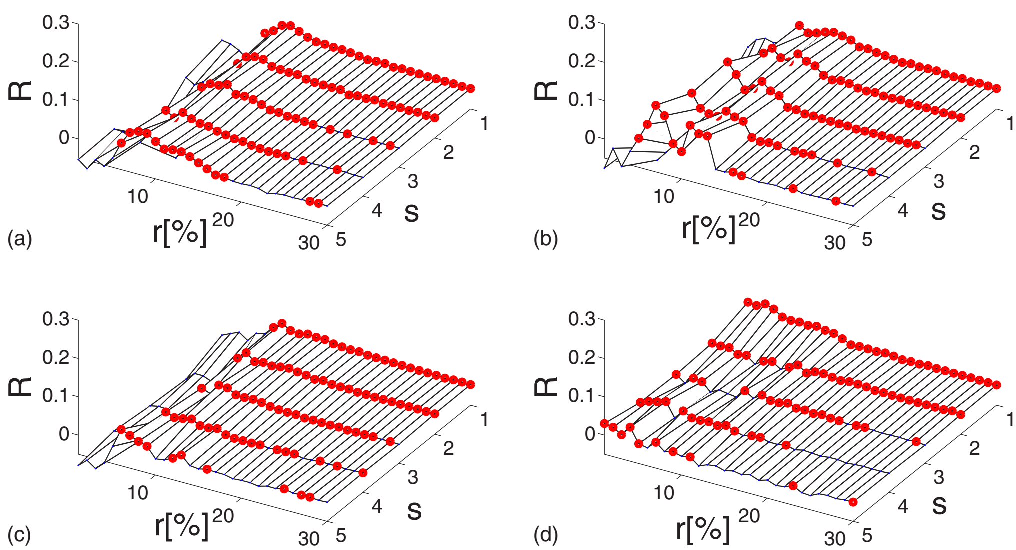

FIG. 6. (Color online) Properties of the improvement ratio R to the variables of r and s for the Ikeda map that is corrupted by observational noise with different noise levels: (a) noiseless and R_{sn}= (b) 40, (c) 30, and (d) =20 dB. Improvements by the proposed method are indicated by closed circles. The data length N is fixed at 500.

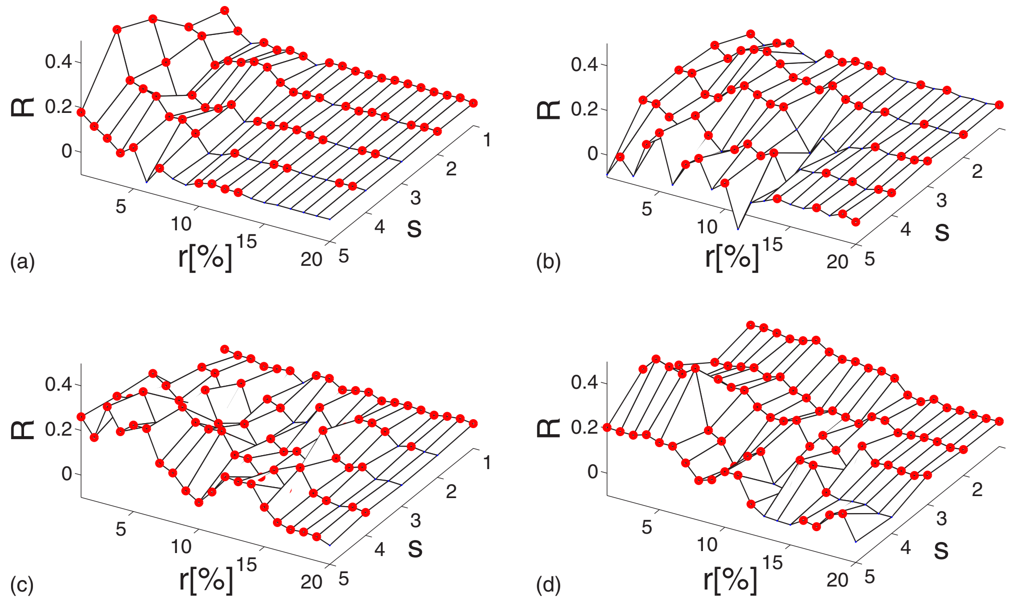

FIG. 7. (Color online) Properties of the improvement ratio R to the variables of r and s for the Japanese vowel /a/ embedded in a d-dimensional state space: $d=$ (a) 2, (b) 3, (c) 4, and (d) 5. Improvements by the proposed method are indicated by closed circles. The data length N is 1000.

VI. CONCLUSIONS

In this paper, we proposed a nonlinear prediction method combining the conventional local linear prediction algorithms with the bootstrap method [12]. Then, we applied the proposed method to the Ikeda map [15] and several real time series [6,9]. As a result, the proposed method is effective even if the near-neighbor radius is small or the time-series data are corrupted by large observational noise. That is, the bootstrap samples can compensate for the lack of local information due to small data lengths or observational noise to estimate an accurate local linear predictor. The proposed prediction method is a powerful tool for nonlinear prediction.

Although we have applied the simple framework of a bootstrap resampling procedure to build better predictors, the framework might be a close relation to nonlinear model selection in statistics [24]. It is also an important future task to discuss the relation not only from a nonlinear dynamical but also from a statistical point of view. In this study, we improved the nonlinear predictability of local linear prediction

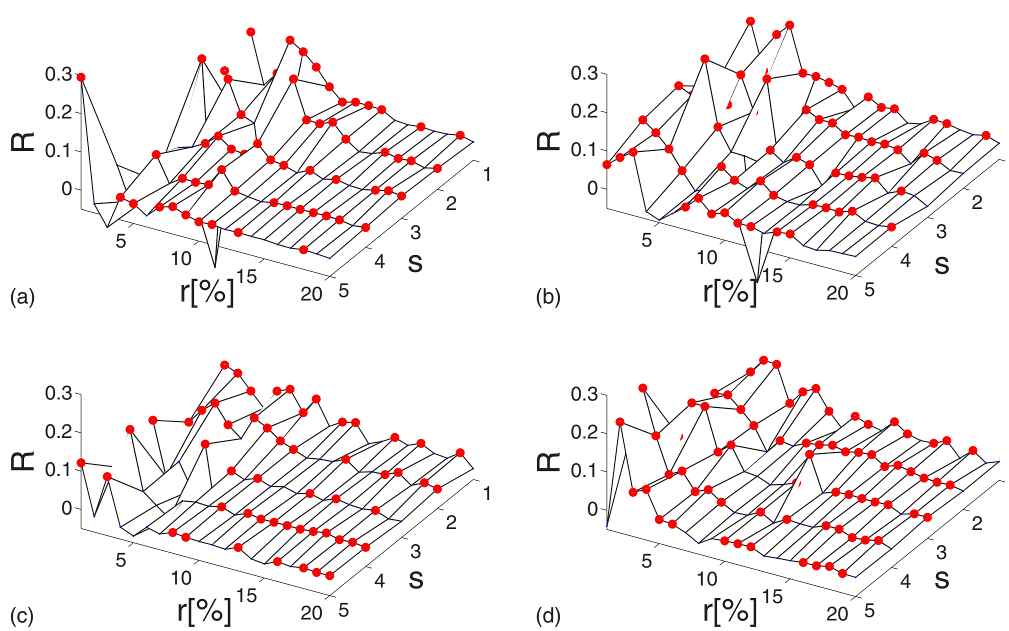

FIG. 8. (Color online) Properties of the improvement ratio R to the variables of r and s for the difference of the number of measles patients embedded in a d-dimensional state space: $d=$ (a) 2, (b) 3, (c) 4, and (d) 5. Improvements by the proposed method are indicated by closed circles.

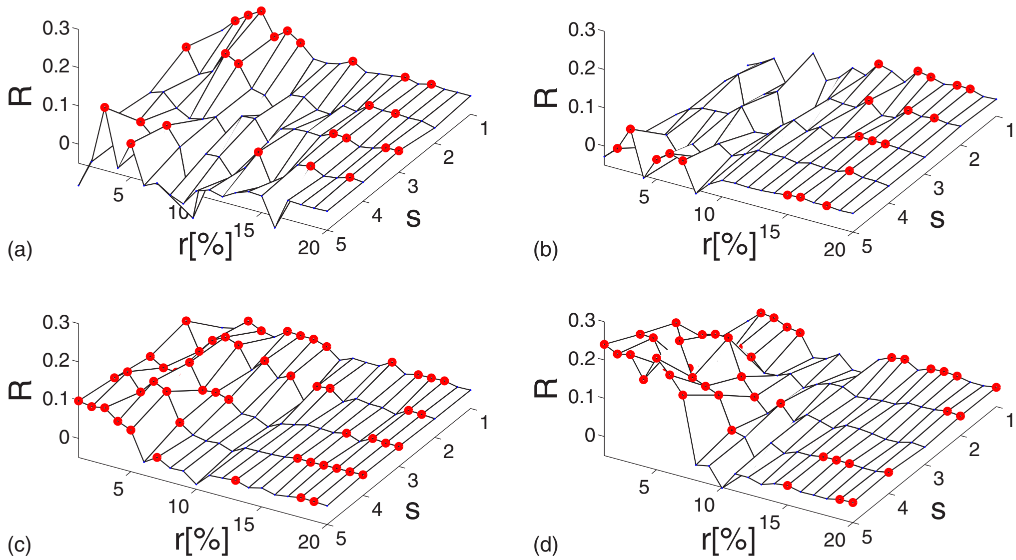

FIG. 9. (Color online) Properties of the improvement ratio R to the variables of r and s for the difference of the number of chickenpox patients embedded in a d-dimensional state space: d= (a) 2, (b) 3, (c) 4, and (d) 5. Improvements by the proposed method are indicated by closed circles.

algorithms. One of the important next steps is to apply the proposed method to the local nonlinear prediction algorithm [4]. In addition, one of the possible extensions of the present framework is to evaluate prediction regions [23]. The application of this framework to relatively high-dimensional data is one of the most important issues in nonlinear time-series analysis.

Moreover, in this paper, we set the resample time as the total number of near neighbors M, because we simply followed the original resampling scheme of the bootstrap method [12]. However, it is also important to evaluate the proposed method with a smaller resample size to reduce the computational load of the proposed method.

ACKNOWLEDGMENTS

We would like to thank Professor Y. Horio, Professor M. Adachi (Tokyo Denki University), Professor M. Small (Hong Kong Polytechnic University), and Dr. K. Hiraoka (Saitama University) for their valuable comments and discussions. Moreover, we are grateful to Dr. G. Sugihara for providing his measles and chickenpox data series. The research of T.I. was partially supported by a Grant-in-Aid for Scientific Research (C) (No. 17500136) from JSPS and a grant from the Research Grants Council of Hong Kong (Project No. PolyU 5235/03E).

[1] E. N. Lorenz, J. Atmos. Sci. **26**, 636 (1969).
[2] M. Sano and Y. Sawada, Phys. Rev. Lett. **55**, 1082 (1985).
[3] J. D. Farmer and J. J. Sidorowich, Phys. Rev. Lett. **59**, 845 (1987).
[4] K. Briggs, Phys. Lett. A **151**, 27 (1990).
[5] J. P. Eckmann, S. O. Kamphorst, D. Ruelle, and S. Ciliberto, Phys. Rev. A **34**, 4971 (1986).
[6] G. Sugihara and R. M. May, Nature (London) **344**, 734 (1990).
[7] M. Casdagli, Physica D **35**, 335 (1989).
[8] K. Judd and A. Mees, Physica D **82**, 426 (1995).
[9] I. Tokuda, R. Tokunaga, and K. Aihara, Int. J. Bifurcation Chaos Appl. Sci. Eng. **6**, 149 (1996).
[10] I. Tsuda, T. Tahara, and H. Iwanaga, Int. J. Bifurcation Chaos Appl. Sci. Eng., Vol. **2**, pp. 313–324, 1992.
[11] P. Bryant, R. Brown, and H. D. I. Abarbanel, Phys. Rev. Lett. **65**, 1523 (1990).
[12] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap* (Chapman and Hall, New York, 1993).
[13] M. Grigoletto, Int. J. Forecast. **14**, 447 (1998).
[14] Y. H. Lee and T. H. Fan, Appl. Econom. Lett. **13**, 41 (2006).
[15] K. Ikeda, Opt. Commun. **30**, 257 (1979).
[16] T. Ikeguchi and K. Aihara, IEICE Trans. Fundamentals **E78-A**, 1291 (1995).
[17] F. Takens, in *Dynamical Systems of Turbulence*, edited by D. A. Rand and B. S. Young, Lecture Notes in Mathematics Vol. 808 (Springer-Verlag, Berlin, 1981), pp. 366–381.
[18] T. Ikeguchi and K. Aihara, J. Intell. Fuzzy Syst. **5**, 33 (1997).
[19] T. Ifukube, M. Hashiba, and J. Matsushima, J. Acoust. Soc. Jpn. **47**, 903 (1991).
[20] T. Miyano, A. Nagami, I. Tokuda, and K. Aihara, Int. J. Bifurcation Chaos Appl. Sci. Eng. **10**, 1973 (2000).
[21] To evaluate the prediction accuracy of the Ikeda map, we use the following measure:

$$E = \frac{\sqrt{[x(t)-\hat{x}(t)]^2 + [y(t)-\hat{y}(t)]^2}}{\sqrt{\sigma_x^2 + \sigma_y^2}},$$

where $\hat{x}(t)$ and $\hat{y}(t)$ are the predicted future values of $x(t)$ and $y(t)$, and σ_x and σ_y are the standard deviations of time series $x(t)$ and $y(t)$, respectively.

[22] We embedded the time series in the d-dimensional state space by Takens' method [17] for real time-series prediction.

[23] D. Haraki, T. Suzuki, and T. Ikeguchi, in *Intelligent Data Engineering and Automated Learning - IDEAL 2006*, edited by E. Corchado, H. Yin, V. Botti, and C. Gyfe, Lecture Notes in Computer Science Vol 4224 (Springer-Verlag, Berlin, 2006), pp. 155–162.

[24] H. Li and G. S. Maddala, Econometric Review **15**, 115 (1996).