Neural Comput & Applic (2016) 27:1129–1141
DOI 10.1007/s00521-015-1930-z

Improving the accuracy of a flood forecasting model by means of machine learning and chaos theory

A case study involving a real wireless sensor network deployment in Brazil

Gustavo Furquim[1] · Gustavo Pessin[2] · Bruno S. Faiçal[1] · Eduardo M. Mendiondo[3] · Jó Ueyama[1]

Received: 13 December 2014 / Accepted: 28 May 2015 / Published online: 14 June 2015

Abstract Monitoring natural environments is a challenging task on account of their hostile features. The use of wireless sensor networks (WSNs) for data collection is a feasible method since these domains lack any infrastructure. However, further studies are required to handle the data collected for a better modeling of behavior and thus make it possible to forecast impending disasters. In light of this, in this paper an analysis is conducted on the use of data gathered from urban rivers to forecast flooding with a view to reducing the damage it causes. The data were collected by means of a WSN in São Carlos, São Paulo State, Brazil, which gathered and processed data about the river level and rainfall by means of machine learning techniques and employing chaos theory to model the time series; this meant that the inputs of the machine learning technique were the time series gathered by the WSN modeled on the basis of the immersion theorem. The WSNs were deployed by our group in the city of São Carlos where there have been serious problems caused by floods. After the data interdependence had been established by the immersion theorem, the artificial neural networks were investigated to determine their degree of accuracy in the forecasting models.

Keywords Wireless sensor network · Machine learning · Time series analysis · Chaos theory · Modeling · Prediction

1 Introduction

Natural disasters, such as landslides, serious floods, fires, volcanic eruptions and the damage they cause, are global problems, which incur a heavy cost in terms of human lives and financial losses. Every year about 102 million people around the world are affected by the problem of flooding, and it is expected that this number will increase in the years ahead. The regions that suffer most from floods are developing countries and urban areas [1]. These are the characteristics of São Carlos in Brazil, where the local climate has changed in the last few years, resulting in serious flooding [2, 3].

Wireless sensor networks are a useful means of carrying out the monitoring of urban rivers and other natural environments since they have a number of attractive features: low costs, particularly with regard to infrastructure, low energy consumption, the fact that they allow access to inhospitable surroundings and are simple to install and their use of high-precision sensors which are adaptable to changes in the environment [4, 5]. A WSN called REDE has been constructed and deployed by our group in the town of São Carlos. This was developed by the Institute of Mathematical Sciences and Computing, University of São

✉ Gustavo Pessin
gustavo.pessin@itv.org

Gustavo Furquim
gafurquim@usp.br

Bruno S. Faiçal
bsfaical@usp.br

Eduardo M. Mendiondo
emm@sc.usp.br

Jó Ueyama
joueyama@icmc.usp.br

[1] Institute of Mathematics and Computer Science (ICMC), University of São Paulo (USP), São Carlos, SP, Brazil

[2] Applied Computing Laboratory, Vale Institute of Technology, Belém, PA, Brasil

[3] São Carlos School of Engineering (EESC), University of São Paulo (USP), São Carlos, SP, Brazil

Paulo (USP), and integrates the e-NOE project (WSN for monitoring urban rivers) which is designed to carry out the monitoring of urban rivers [5, 6]. Until now, our WSN system has only been capable of detecting floods when they have already taken place. Hence, we are now keen to enable it to predict floods before they occur, and one of the aims of this paper is to explore this area and thus ensure that the people at risk are evacuated in time. It involves making measurements that can help us to reduce a wide range of problems arising from floods.

Most of the nodes of the REDE system (REde de sensors sem fio para Detectar Enchentes—WSN for flood detection) provide measurements of the water pressure at the bottom of the river by converting this value into centimeters to describe the river level; in addition there is one node that provides information about the volume of rainfall. These values are collected at regular time intervals and can thus enable the level of the river to be known over a period of time. In this way, the gathered data can be interpreted as a time series, which allows us to study, model and investigate the question of behavioral prediction. A simple definition of a time series is that it is an orderly sequence of observations collected at regular intervals [7, 8]. This means that the data collected by the REDE system constitutes a time series which can be studied in the light of the concepts of time series analysis.

This paper is based on the assumption that there is a temporal relationship between several different observations of the level of the river and hence the value of an observation at a particular time depends to some extent on past values. For this reason, we seek to determine the temporal relationship, as well as to model and forecast the level of the rivers. To achieve this, we evaluated techniques that originate from the time series analysis or, more precisely, the sub-area of chaos theory. The employment of concepts of chaos theory enables us to interpret the time series in a non-trivial way. Our paper [9] has shown that treating a time series in a varied way (i.e., using data characteristics rather than raw values) could improve the accuracy of machine learning techniques. Our reason for employing chaos theory techniques [7, 8, 10] is that they enable the time series to be modeled in a varied way and can thus improve the accuracy of machine learning.

There is a theorem, put forward by Takens [11], known as the embedding theorem that states that a time series can be reconstructed in vectors with m values (described as the embedding dimension). Each of these values corresponds to an observation that is spaced out in intervals, in accordance with a time delay (or separation dimension) called τ. As several studies have shown, these vectors represent the interdependent relationships between observations, and since they increase the accuracy of the modeling, they also improve the accuracy of the prediction. After the time series is modeled by means of Takens' embedding theorem [11], machine learning techniques such as artificial neural networks (ANN), support vector machines (SVM) or even a combination of several techniques can be used to create forecast models. More details on chaos theory and how it is employed in our work are given in Sects. 3 and 4.

This paper extends [12] by using a more comprehensive dataset, which includes a pluviometer sensor. This pluviometer sensor was deployed in the REDE system to improve the accuracy of the forecasting model, and its deployment allows a comparative analysis to be made with the previous results obtained from [12]. In this paper, we also evaluate the use of the Elman recurrent neural network for time series forecasting and the results obtained are compared with the multilayer perceptron. In addition to the data collected and studied in March 2013, [12], use is also made of the data collected during the entire month of April 2014 when there was serious flood in São Carlos, with river levels at nearly 300 cm, which surpassed the safety threshold level.

The remainder of this paper is divided as follows: Sect. 2 examines some related studies that employ the chaos theory and WSNs or are concerned with flood forecasting. In Sect. 3, the tools of the chaos theory and the immersion theorem are explained in greater detail. Section 4 outlines the proposed method and describes the WSN that is used, as well as the data handling and the artificial neural network. This work ends with an analysis of the results obtained (Sect. 5) followed by the conclusion and suggestions for future work (Sect. 6).

2 Related work

Seal et al. [4] outline a flood forecasting scheme using a hybrid approach (centralized and distributed) for WSNs in rivers. The WSN architecture consists of several sensors for collecting data combined with processor nodes, where the forecast algorithm is implemented. It also has centers of manually operated monitoring, which implement redundancy by comparing the real situation with the forecast and initiate evacuation procedures. The data flow basically comprises: data collection, the calculation of the coefficient of regression, updating coefficients of regression, sending the results and then informing the community. The forecast model uses robust linear regression, since it does not depend on the number of parameters. In addition to the results, they also show other related studies and make clear the applicability of WSNs in collecting data for flood forecasting.

In [13] an architecture is proposed for slope disaster prediction and monitoring. This architecture is based on WSNs and mobile communication that transmit warnings and indicate the areas at risk of disasters. The main elements of the architecture are: (1) mobile user site that implements the user interface, (2) hillslopes monitoring sensor site for the physical implementation of the WSN, (3) integrated service server, which provides services like network integration, and (4) intelligent hillslopes decision system, which predicts the degree of risk of the hillslopes. The analytic network process (ANP) model is used to predict the degree of risk; this has an accuracy of 88.33%.

Ishii and Mello [7] propose an online prediction approach to support data access optimization in distributed systems. The data acquisition was built using the Unix DLSym library and transformed into a multidimensional time series. The time series was then analyzed to evaluate the best model for making predictions. As a result, the chaos theory tools were used to unfold the data and the radial basis function (RBF) neural network was used to model the time series.

3 Concepts of the chaos theory

Takens [11] observed that a time series $x_0, x_1, \ldots, x_{n-1}$ can be reconstructed (i.e., modeled) in a multidimensional space $x_n(m, \tau) = (x_n, x_{n+\tau}, \ldots, x_{n+(m-1)\tau})$, also called time-delay coordinate space, where m is the embedded dimension and τ represents the time delay (or separation dimension). This mapping (or reconstruction) technique allows transforming dynamical system observations in a set of points in an m-dimensional Euclidean space. This reconstruction supports the acquisition of dynamical systems rules and can thus simplify the study of behaviors and their applications under different circumstances, such as the study of orbits, trends and prediction [14].

To understand the embedded (number of dimensions) and separation (time delay) dimensions, consider the logistic map outputs that are reconstructed in a multidimensional space where $m = 2$ and $\tau = 1$, which results in the pairs of points (x_t, x_{t+1}) (Fig. 1b). After the reconstruction, the logistic map, which behaved as a random walk (Fig. 1a), can be studied and modeled in a simpler way.

The embedded dimension defines the number of axes for the time-delay coordinate space. This determines the number of dimensions necessary to unfold the reconstructed series. In this case, the series required two dimensions, but others may need more. This behavior is, for example, observed in the Lorenz attractor which requires three dimensions [15]. Apart from the embedded dimension, there is still the separation one, which supports the extraction of the periodical behavior patterns of the series. This dimension gives information about the time delay of historical observations. The embedded and separation dimensions allow the modeling of the series, although we need to find these dimensions the series, including the ones generated by experimental data. According to Fraser and Swinney [16], the auto-mutual information technique (AMI) achieves better results when estimating the separation dimension. The series separation can be obtained by employing AMI under different time delays. Following this, one plots a curve in accordance with time delays (starting at 1 and incrementing) and the first minimum is considered to be the main candidate for the separation dimension [10]. As well as determining whether it is a good candidate, one should also consider other separation dimension values and observe the characteristics of the time-delay coordinate space that is obtained.

After defining the separation dimension, we must find the embedded one. Kennel [17] proposed the false nearest neighbors method (FNN) to estimate the embedded dimension. This method computes the neighbors closest to every data point in the time-delay coordinate space, starting with embedded dimension equals 1. After this, a new dimension is added and the distance between the closest neighbors is calculated again. When this distance increases, the data points are considered false nearest neighbors, which makes it clear that there is a need for more dimensions to unfold the behavior of the series [10]. After defining both dimensions, the embedded theorem is employed, as previously shown, where the time series $x_0, x_1, \ldots, x_{n-1}$ is reconstructed in a multidimensional space or time-delay coordinate space $x_n(m, \tau) = (x_n, x_{n+\tau}, \ldots, x_{n+(m-1)\tau})$ where the component m represents the embedded dimension, that is, the number of dimensions to unfold the series, and τ is the separation, that is, the time delay to consider historical observations. The reconstruction unfolds the dynamical system, which allows the rule to be obtained. After this unfolding, different approaches can be adopted to model the system, ranging from artificial neural networks to numerical methods.

4 Methods of flood forecasting

As stated in the previous sections, our main concern is to investigate how the use of the chaos theory can help in improving the accuracy of machine learning techniques employed in forecasting, and also how to employ the data gathered from rivers by means of WSNs. The WSN deployed in the town of São Carlos, Brazil (Fig. 2), comprises the following: (1) Sensors 1, 2 and 3 (shown as yellow dots) that give the measurements of the pressure at the bottom of the river and convert this value into the

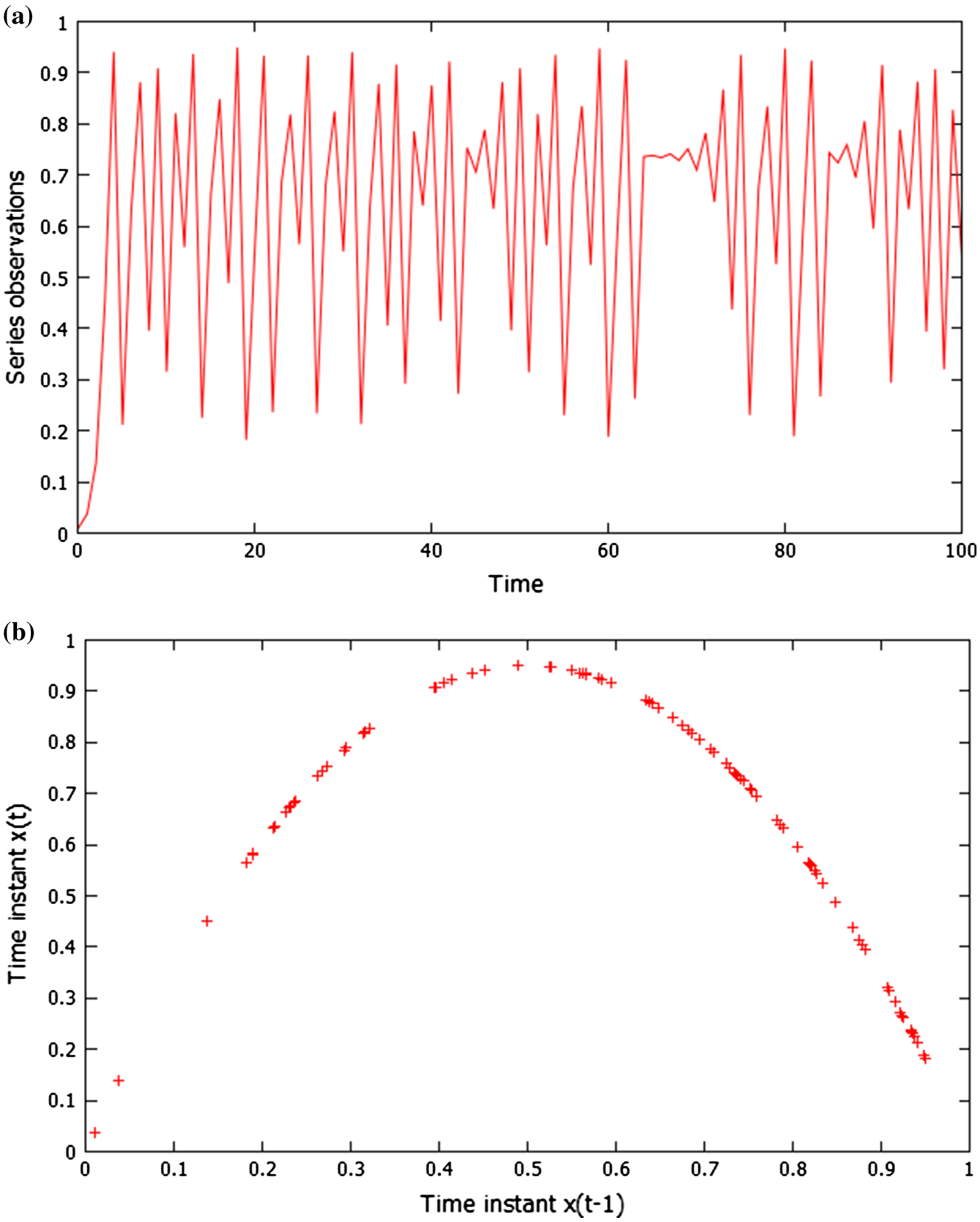


Fig. 1 **a** Logistic map outputs—first 100 observations and **b** observations of the reconstructed logistic map (embedded dimension 2 and separation 1)

height in centimeters, (2) a pluviometer (shown as light blue—installed near the router) that gives measurements of the rainfall volume in millimeters, (3) a router (shown as a light blue dot) that is able to increase the communication range and allow the sensors to be positioned in a wider area, and (4) a base station (shown as a red dot) that gathers the information from several sensors and allows a more effective analysis of the data to be conducted. In addition to measuring the height of the river level, Sensor 3 is equipped with a photographic camera which makes visual information available about the river conditions. In this paper, data were collected by Sensor 1 during a period of two months: (1) March 2013 (from March 1, 2013, to March 31, 2013), at intervals of 5 minutes between the measurements, and (2) April 2014 (from April 1, 2014 to April 30, 2014), at intervals of 1 min between the measurements. The behavior of the river level in March 2013 is illustrated in Fig. 3a, and the behavior of the river in April 2014 is illustrated in Fig. 3b, where time is described in a days:hours format. The rainfall pattern in April 2014 is illustrated in Fig. 3c. The required outcome (following the data handling and modeling of the system through the multilayer perceptron artificial neural network and the Elman recurrent neural network) is the level of the river in the next instant.

Before employing the proposed approaches, it is necessary to adopt a checking procedure to find out whether or not the times series has a chaotic behavior. The Lyapunov

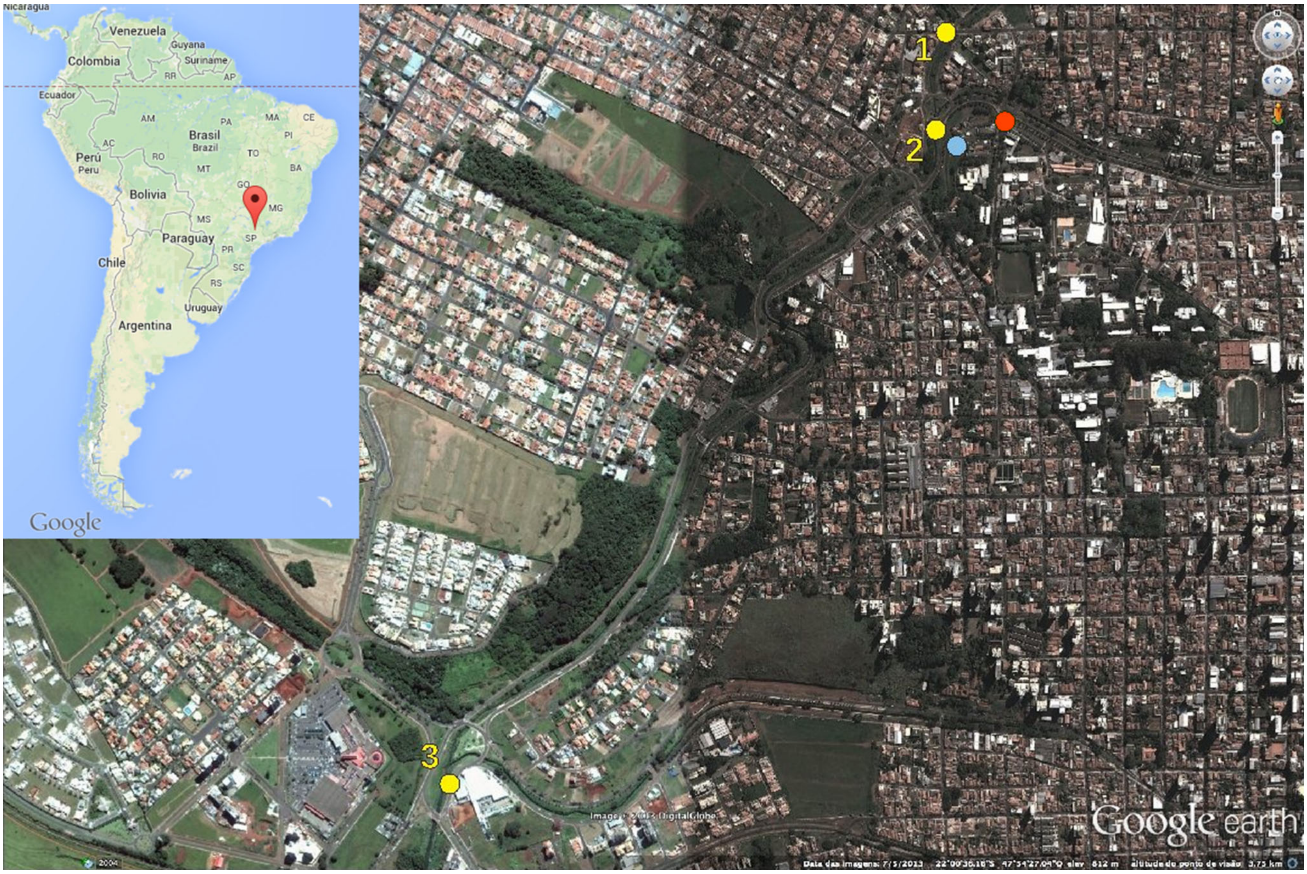


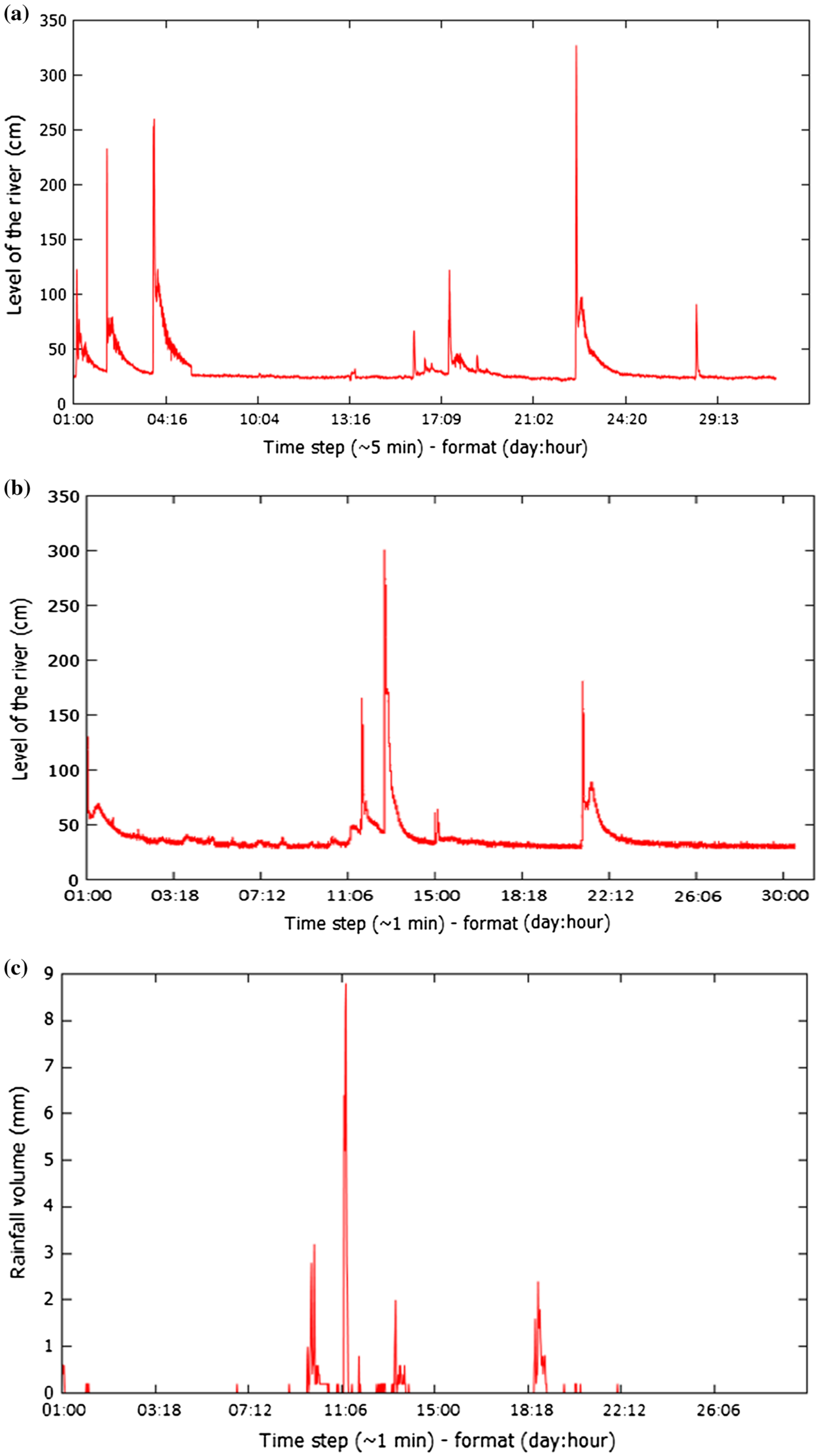
Fig. 2 Positioning of the sensors (*yellow dots*), pluviometer (*light blue dot*—installed near the router), base station (*red dot*) and router (*light blue dot*) at São Carlos, SP, Brazil [9] (color figure online)

exponent was employed to carry this out. Figure 4a, b shows the behavior of the Lyapunov exponent for the time series that represents the river level. As can be seen, there is a scaling region in both graphs, so the estimated maximum Lyapunov exponent (λ) is the curve slope plotted in the region [18]. The estimated values for the maximum Lyapunov exponent are as follows: $\lambda = 0.153$ for March's river level time series and $\lambda = 0.028$ for April's river level time series. As there is $\lambda > 0$ in both cases, it can be stated that both time series have a chaotic behavior [8].

When this approach is adopted, the first stage is to calculate the auto-mutual information technique (AMI) to determine the value of the separation dimension (time delay τ) for the river level in March 2013. As proposed in [19], the first minimum of this graph from left to right should be selected as the separation dimension, which is $\tau = 21$. However, as observed in other studies [10, 17], since there is a slight reduction in AMI as the separation dimension increases, a good estimate for the separation dimension would be $\tau = 1$. In this work, both values for τ are considered which can provide different immersions and an attempt is made to determine which is the best. After computing the separation dimension, the next stage is to estimate the embedding dimension (m) through the false nearest neighbors (FNN) method [17]. If this method is employed, the embedding dimension is selected when the fraction of false neighbors is equal to zero; however, this is very hard to obtain for real-world data, as they may be affected by noise. In noisy scenarios, the number of dimensions can be selected when the fraction of false neighbors falls to less than 30 % as shown in [20]. Figure 7 shows the results of FNN when considering $\tau = 1$ (Fig. 7a) and $\tau = 21$ (Fig. 7b), in which the x-axis corresponds to embedding dimensions and the y-axis is related to the fraction of false neighbors. Thus, we selected $m = 4$ when $\tau = 1$ and $m = 11$ when $\tau = 21$. Greater values can be considered for the embedding dimension, although according to Kennel [17], this will not have much influence because once the behavior of the time series has been disclosed, it can be understood and studied.

Figure 5 shows the different stages that were followed when employing chaos theory and machine learning in flood forecasting. Figure 6 displays the input and output values used in the experiments by considering $m = 4$ and $\tau = 1$ (Fig. 6a) and $m = 11$ and $\tau = 21$ (Fig. 6b). The points in orange represent the values that act as inputs of

Fig. 3 **a** Level of the river during March 2013. **b** Level of the river during April 2014. **c** Rainfall volume during April 2014



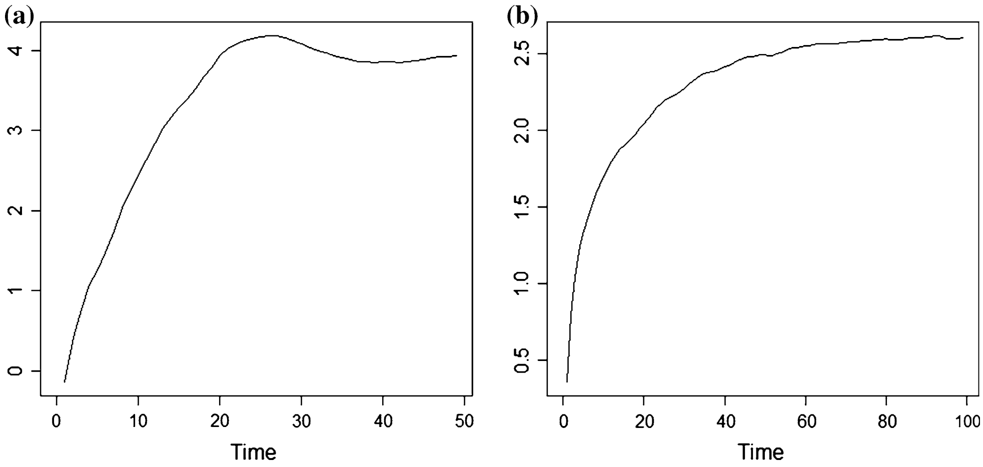


Fig. 4 **a** Estimating the maximal Lyapunov exponent of the March 2013 river level. **b** Estimating the maximal Lyapunov exponent of the April 2014 river level

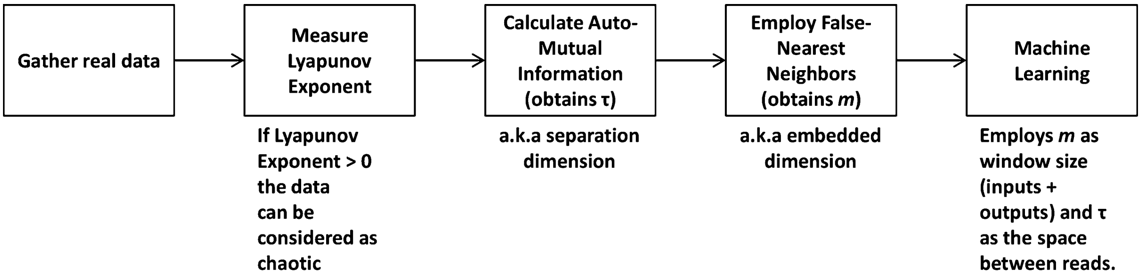

Gather real data
Measure Lyapunov Exponent
Calculate Auto-Mutual Information (obtains τ)
Employ False-Nearest Neighbors (obtains m)
Machine Learning
If Lyapunov Exponent > 0 the data can be considered as chaotic
a.k.a separation dimension
a.k.a embedded dimension
Employs m as window size (inputs + outputs) and τ as the space between reads.

Fig. 5 Our steps for employing chaos theory and machine learning in flood forecasting (see Fig. 6 for additional information)

the machine learning technique; the points in red represent values that act as outputs. These diagrams (Fig. 6a, b) show a generic time series, i.e., they do not represent real data gathered by the WSN. They seek to illustrate, in a more understandable way, how the values of m and τ, obtained by the chaos theory, can be interpreted as points in a generic time series. In practical terms, m means the whole window size (considering inputs and outputs) and τ means the space between reads to be considered (Fig. 7).

After employing the concepts of the chaos theory to obtain the separation and the embedding dimensions (see Figs. 5, 6), we evaluated how two machine learning techniques carry out the forecasting. The evaluated techniques are multilayer perceptron (MLP) and Elman recurrent neural networks (E-RNN). The Waikato Environment for Knowledge Analysis (WEKA) [21] was employed to model and evaluate both techniques. The results are obtained by employing a tenfold cross-validation procedure. It should be noted that the embedding dimension represents the full window size (inputs + outputs). Hence, the inputs of the NNs consist of the size of the embedding dimension minus one. The first results (shown in Sect. 5) regard the river level as inputs; the ensuing results take into account both the river level and rainfall.

The same measures were applied to the river level in April 2014. The values for the separation dimension that considered the AMI technique were $\tau = 24$ and $\tau = 1$. Figure 8 shows the results of FNN when considering $\tau = 1$ (Fig. 8a) and $\tau = 24$ (Fig. 8b). Again, the best results were obtained with $\tau = 1$ and $m = 4$, followed by $\tau = 1$ and $m = 3$; these values differ from former estimates. Hence, when employing real-world data in a time series, values near to the values obtained by AMI and FNN must be taken into account to model the time series.

5 Results and discussion

Since there were different estimated values for the separation and embedding dimensions for each time series, the first experiments were conducted with a combination of m and τ and attempted to determine which set could provide the more accurate results. Tables 1 and 2 show the mean absolute error (MAE), root mean square error (RMSE) and coefficient of determination (R^2) obtained from a combination of m and τ. For both series (March and April), the best results were obtained when employing $m = 4$ and $\tau = 1$. By means of these investigations, it can be observed that, since the datasets have been obtained from a real environment that is subject to noise, the auto-mutual information provides good values for the separation

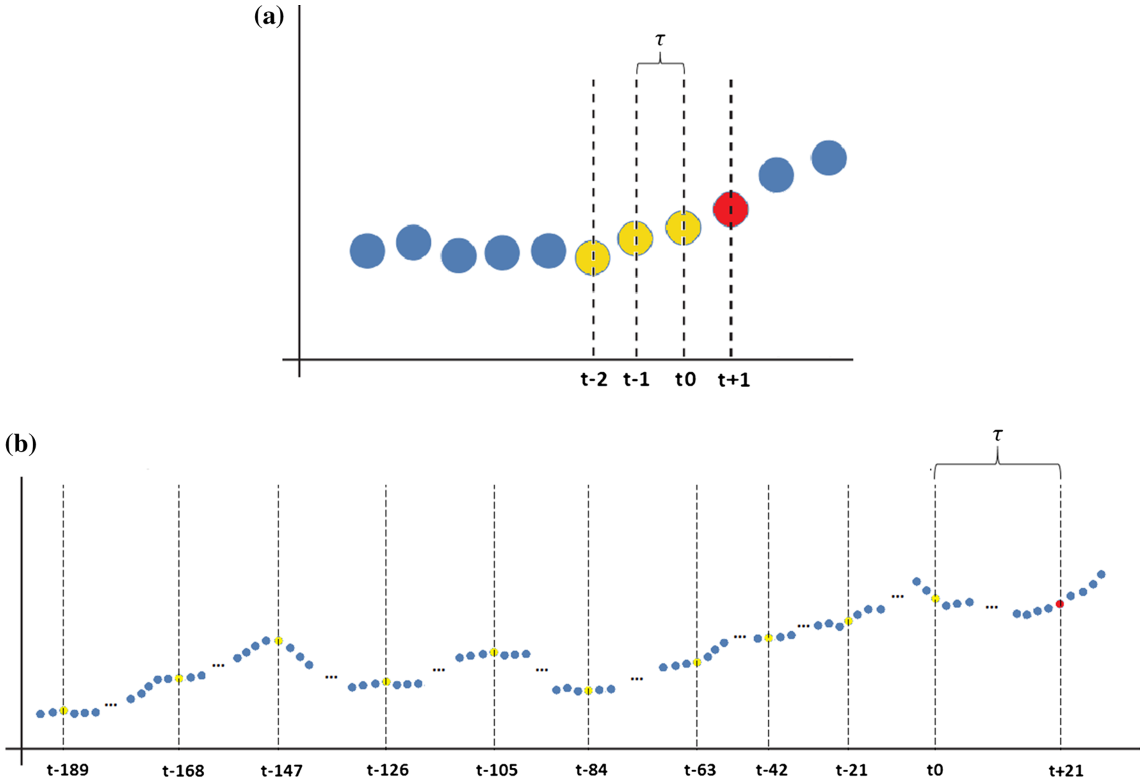


Fig. 6 Representation of the input and output values used in the experiments by considering **a** $m = 4$ and $\tau = 1$ and **b** $m = 11$ and $\tau = 21$. The points in *orange* represent the values to be employed as inputs of the machine learning technique. The points in *red* represent output values. These diagrams display a generic time series, i.e., they do not represent real data gathered by the WSN. Its aim is to show in a more understandable way how the values of m and τ, obtained by the chaos theory, can be interpreted as points in a generic time series. In practical terms, m means the whole window size (considering inputs and outputs) and τ means the space between reads to be considered (color figure online)

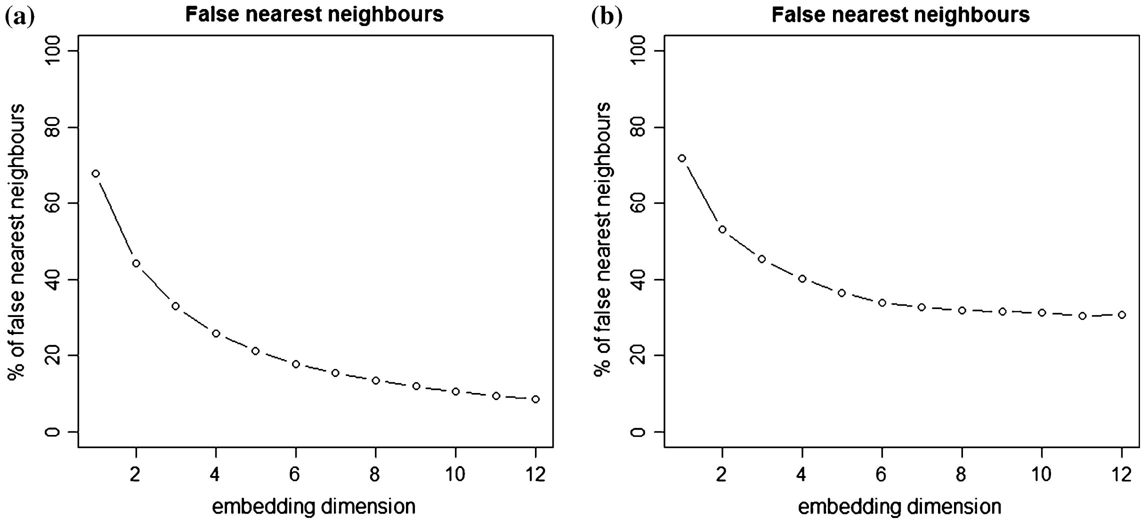


Fig. 7 Percentage of false neighbors for the river level in March 2013, considering **a** $\tau = 1$ and **b** $\tau = 21$

dimension. However, other minimum points (and the value of $\tau = 1$) must be regarded as alternative choices.

Figures 9 and 10 show the behavior of the river during the month of March 2014 (red line) and the values predicted by the MLP (green line). It can be seen that in Fig. 9 where the values for $\tau = 1$ and $m = 4$ are shown, the lines are usually overlapping, which suggests that the predicted values are as close as expected. In Fig. 10, in

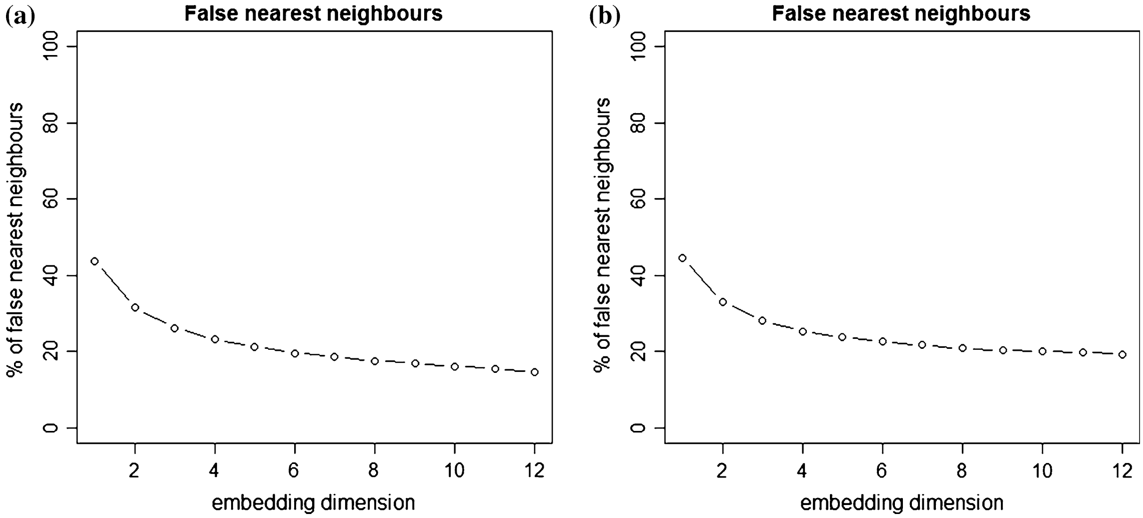


Fig. 8 Percentage of false neighbors for the river level in April 2014, considering **a** $\tau = 1$ and **b** $\tau = 24$

Table 1 Forecasting results (MAE, RMSE and R^2) while employing the MLP that takes into account river level of March 2013

Separation dimension (τ)	Embedded dimension (m)	MAE	RMSE	R^2
1	4	1.573*	3.838*	0.961*
21	1	9.476	20.069	0.168
1	3	1.895	4.154	0.954
24	3	3.848	15.922	0.327

* More accurate results

Table 2 Forecasting results (MAE, RMSE and R^2) while employing the MLP that takes into account river level of April 2014

Separation dimension (τ)	Embedded dimension (m)	MAE	RMSE	R^2
1	4	1.105*	1.402*	0.994*
21	11	1.629	4.972	0.929
1	3	1.118	1.532	0.993
24	3	2.265	6.139	0.892

* More accurate results

which $\tau = 11$ with $m = 21$ was employed, the lines end up by diverging, particularly at the peaks which are important regions because they show the instants when there is a risk of flooding. Figure 11 shows the behavior of the river during April 2014 for the best τ and m.

The second stage in the evaluations includes information about rainfall, gathered by a rain gauge deployed in the WSN. We maintained the values of τ and m that were obtained in previous evaluation (that took account of the river level). Table 3 shows the relationship between the river level and the amount of rainfall during a peak period in April 2014. As can be seen in the Table, when there is a time t_0 in the instant where the pluviometer showed some rainfall, the river level appears to be influenced by it in the next time stage (t_1); however, it changes when we observe other parts of the series. In the light of this, it is hard to determine when and for how long the rain influences the river level; these questions are related to the characteristics of the basin, its environment and seasons. Another problem is that since there is only one pluviometer, it might not be possible to sense rain in the observed river (since it might be raining in its tributaries).

Evaluations were carried out with a window size that ranged from one to ten readings of the pluviometer (i.e., the window size of the rainfall). It should be pointed out that the larger the window size, the longer the time (memory), since a window with a size equal to 1 represents the current sensed data, a window size with a size equal to 2 represents the current and the former sensed data, and so on. In this second experiment, two machine learning approaches were compared: MLP and E-RNN. We decided to compare the MLP with the E-RNN because of its recurring characteristics and the fact that the exit of the neurons is used as the entry for other neurons. This makes the E-RNN a good

Fig. 9 Expected and predicted river levels obtained by the MLP while employing the best set of embedded and separation dimensions found ($\tau = 1$ and $m = 4$) for March 2014

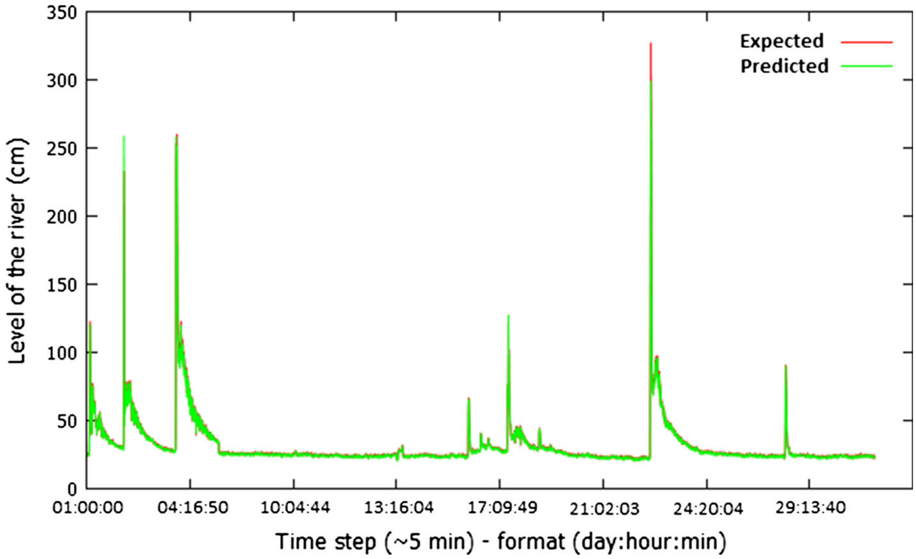


Fig. 10 Expected and predicted river levels obtained by the MLP while employing $\tau = 11$ and $m = 21$ for March 2013. The lines end up being divergent, particularly where there are sudden changes (most times near the *peaks*)

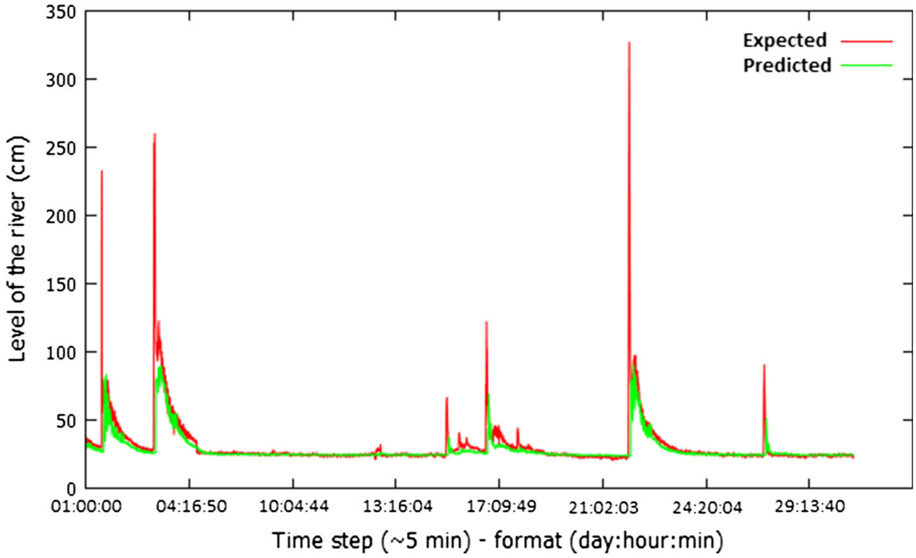
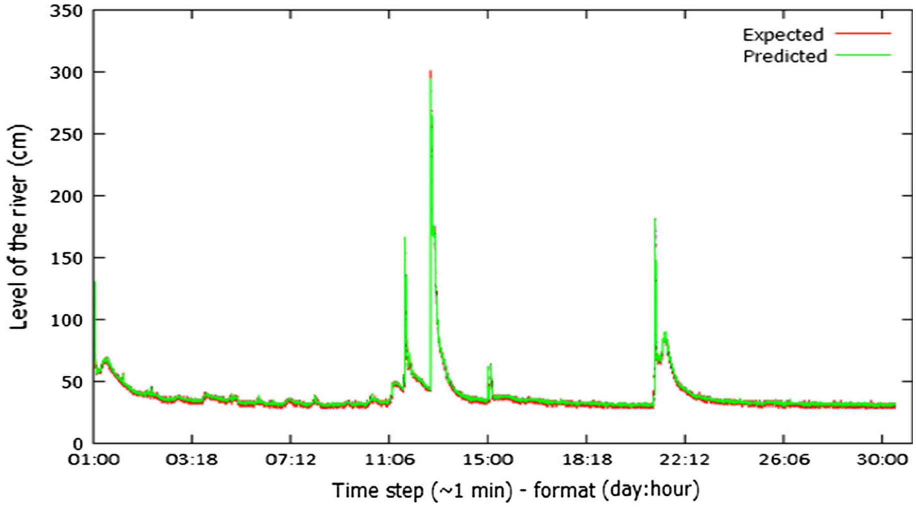


Fig. 11 Expected and predicted river levels obtained by the MLP while employing the best set of embedded and separation dimensions found ($\tau = 1$ and $m = 4$) for April 2014



candidate for the processing of variant systems with regard to time, as is the case of the predictions of the time series.

Table 4 shows the results when the level of the river and the amount of rainfall are taken into account as inputs in the MLP. Table 5 shows the results of the E-RNN. We evaluated three different hidden layer sizes and 10 different rainfall window sizes, plus the configuration without rain. It is clear that in both cases (MLP and E-RNN) there is an improvement in the results when the rainfall information is employed as input together with the river level. In the case of the MLP, the best results were obtained with a window size of seven readings, while the best results for the E-RNN were obtained with a window size of 10 readings. In both cases, the best results were obtained by employing the number of hidden layers as 3× number of inputs.

Table 3 River level and rainfall volume during a peak event on April 13, 2014

Time (hh:mm)	River level (cm)	Rainfall volume (mm)
16:31	23.24	0.0
16:36	23.55	0.0
16:41	23.71	0.0
16:46	22.45	0.0
16:51	23.55	6.4
16:56	36.74	5.8
17:01	75.99	5.2
17:06	100.95	8.2
17:11	139.42	8.0
17:16	200.18	4.8
17:21	231.10	2.8
17:26	242.09	2.4
17:31	244.29	1.4
17:36	236.44	0.0
17:41	238.48	0.0
17:46	217.92	0.0

A comparison between the MLP and the E-RNN shows that the best results were obtained when the MLP was employed (see Tables 4, 5). Figure 12 shows lines with the expected and obtained values (collected from the river level and predicted by the best MLP) for a period in April 2014. Although the MLP corresponds to most of the points, it can be noted that where there are sudden changes (for example, near 17:10 and 17:50), the expected values have the largest errors.

6 Conclusion and plans for future work

In this paper, a study has been carried out on the use of Takens' theorem [9] for the preprocessing of collected data from urban rivers by means of WSNs and subsequent handling of these data for flood prediction by using artificial neural networks. Four different combinations of separation and embedding dimension were evaluated as inputs of an MLP that only took account of the river level. In a second stage, there was an evaluation of whether information about rainfall could improve the results; in addition, we evaluated the E-RNN and compared its results with an MLP. The results show that the MLP performs better than the E-RNN for our data; moreover, the measurement of rainfall allowed us to improve the performance of both MLP and E-RNN.

In future work, we intend to extend our examination of the machine learning techniques so that they can be used

Table 4 Forecasting results (MAE and RMSE) while employing the MLP that takes into account the river level and rainfall in April 2014

Rainfall window size	1× # of inputs		2× # of inputs		3× # of inputs	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
No.	1.0968	1.3933	1.0325	1.3112	1.0527	1.3375
1	1.1035	1.4017	1.0449	1.3304	1.0534	1.3424
2	1.0536	1.3430	1.0521	1.3394	1.0527	1.3443
3	1.0565	1.3474	1.0648	1.3573	1.0614	1.3526
4	1.0812	1.3799	1.0875	1.3923	1.0917	1.3974
5	1.0791	1.3883	1.0956	1.4220	1.1047	1.4266
6	1.0881	1.4114	1.1065	1.4383	1.1126	1.4351
7	0.8983	1.2059	0.8965	1.2043	0.8957*	1.1681*
8	1.0058	1.3292	1.0017	1.3066	1.0038	1.2970
9	1.0449	1.4143	1.0414	1.4067	1.0389	1.3646
10	1.0258	1.3712	1.0206	1.3486	1.0111	1.3134

All the R^2 are larger than 0.994, and the asterisks show the lowest error. With regard to the size of the hidden layer, the best ANN obtained in the previous stages had 3 inputs ($m = 4$). Hence, in these evaluations, 1× number of inputs means that the ANN has 3 neurons in the hidden layer and so on, plus the size of the rainfall window

Table 5 Forecasting results (MAE and RMSE) while employing the E-RNN that takes into account the river level and rainfall in April 2014

Rainfall window size	1× # of inputs		2× # of inputs		3× # of inputs	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
No	1.0357	1.3113	0.9750	1.2433	0.9770	1.2466
1	0.9988	1.2719	0.9827	1.2557	0.9739	1.2459
2	0.9754	1.2456	0.9764	1.2496	0.9744	1.2496
3	0.9961	1.2727	0.9804	1.2581	0.9781	1.2571
4	0.9894	1.2736	0.9847	1.2687	0.9858	1.2710
5	0.9845	1.2728	0.9856	1.2935	0.9845	1.2937
6	0.9858	1.2915	0.9891	1.3029	0.9870	1.2880
7	1.0349	1.3596	1.0402	1.3721	1.0404	1.3577
8	0.9935	1.3005	0.9939	1.3047	0.9875	1.2832
9	0.9965	1.3545	0.9930	1.3485	0.9928	1.3056
10	0.9528	1.2715	0.9533	1.2783	0.9116*	1.2020*

All the R^2 are larger than 0.994, and the asterisks show the lowest error. With regard to the size of the hidden layer, the best ANN obtained in the previous stages had 3 inputs ($m = 4$). Hence, in these evaluations, 1× number of inputs means that the ANN has 3 neurons in the hidden layer and so on, plus the size of the rainfall window

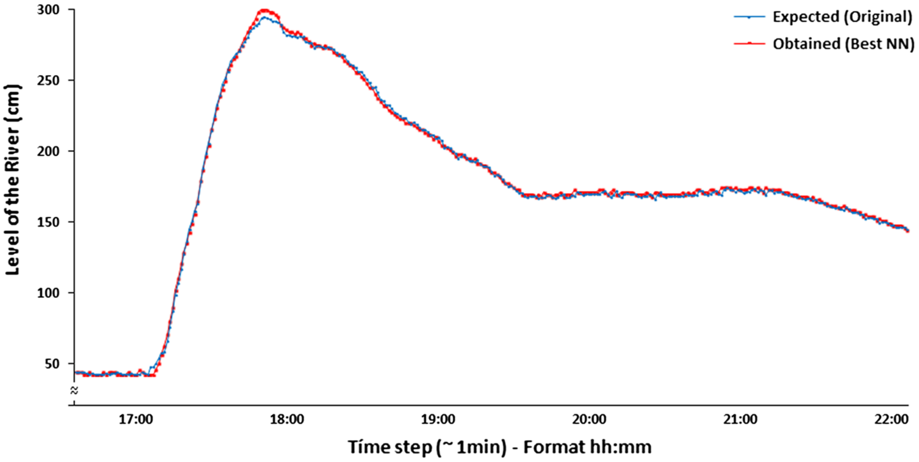


Fig. 12 *Lines* with values that are expected (collected from the river level) and obtained (predicted by the best acquired NN) for a period in April 2014. The best NN is an MLP that takes the river level and rainfall as inputs, as can be seen in Table 4. Although the NN corresponds to most of the points, it should be noted that where there are sudden changes (for example, near 17:10 and 17:50), the expected values have the largest errors

with the preprocessed data to model the system. In addition, although readings from other sensors in the REDE system (Sensor 2 and 3—see Fig. 2) were not used, they might contain information that could improve the accuracy of the forecasting. Another study that should be undertaken is to conduct a statistical analysis of the series with regard to the level of the river and the time series containing the volume of rainfall. An analysis of this kind could determine more clearly the relationship between the different series and the window size that must be used and thus broaden our knowledge of the machine learning technique that is employed. In future, we also seek to embed this prediction model in the sensors and thus make it possible to take action in a more independent way in extreme situations (for example, when there is a breakdown of communication or the destruction of nodes). This kind of integration reduces the energy consumption of each node, and this is a factor that requires further study.

Acknowledgments The authors would like to acknowledge the financial support granted by São Paulo Research Foundation (FAPESP) process IDs 2012/22550-0, 2014/19076-0 and 2008/58161-1. Also, the authors would like to thank Filipe A. N. Verri for his time and fruitful discussions. The third author would like to acknowledge the Capes Foundation, Ministry of Education of Brazil and FAPESP process ID 2013/18859-8. Jó Ueyama would like to thank the Office of Naval Research Global for funding part of his research project.

References

1. Freitas CM, Ximenes EF (2012) Floods and public health—a review of the recent scientific literature on the causes, consequences and responses to prevention and mitigation. Ciência e Saúde Coletiva 17:1601–1616
2. Barbosa RVR, Vecchia FAS (2009) Estudos de Ilha de Calor Urbana por meio de Imagens do Landsat 7 Etm+: Estudo de Caso em São Carlos (SP). Rev Minerva 6(3):273–278
3. Pozza SA (2005) Identificação das Fontes de Poluição Atmosférica na Cidade de São Carlos - SP. Master's thesis, Federal University of Sao Carlos

4. Seal V, Raha A, Maity S, Mitra SK, Mukherjee A, Naskar MK (2012) A simple flood forecasting scheme using wireless sensor networks. Int J Ad hoc Sens Ubiquitous Comput 3:45–60
5. Ueyama J, Hughes D, Man KL, Guan S, Matthys N, Horre W, Michiels S, Huygens C, Joosen W (2010) Applying a multi-paradigm approach to implementing wireless sensor network based river monitoring. First ACIS international symposium on cryptography and network security, data mining and knowledge discovery, E-commerce its applications and embedded systems (CDEE), 2010
6. Hughes D, Ueyama J, Mendiondo E, Matthys N, Horre W, Michiels S, Huygens C, Joosen W, Man K, Guan SU (2011) A middleware platform to support river monitoring using wireless sensor networks. J Braz Comput Soc 17(2):85–102
7. Ishii RP, de Mello RF (2012) An online data access prediction and optimization approach for distributed systems. IEEE Trans Parallel Distrib Syst 23(6):1017–1029
8. Mello RF (2011) Improving the performance and accuracy of time series modeling based on autonomic computing systems. J Ambient Intell Humaniz Comput 2(1):11–33
9. Furquim G, Neto F, Pessin G, Ueyama J, Clara M, Mendiondo EM, Souza P, Dimitrova D, Braun T (2014) Combining wireless sensor networks and machine learning for flash flood nowcasting. 28th International conference on advanced information networking and applications workshops (WAINA), 2014, pp 67–72
10. Mello R, Yang L (2009) Prediction of dynamical, nonlinear, and unstable process behavior. J Supercomput 49:22–41
11. Takens F (1981) Detecting strange attractors in turbulence. Lecture Notes Math 898:366–381
12. Furquim G, Mello R, Pessin G, Faical B, Mendiondo E, Ueyama J (2014) An accurate flood forecasting model using wireless sensor networks and chaos theory: a case study with real WSN deployment in Brazil. Eng Appl Neural Netw Commun Comput Inf Sci 459:92–102
13. Wu CI, Kung HY, Chen CH, Kuo LC (2014) An intelligent slope disaster prediction and monitoring system based on WSN and ANP. Expert Syst Appl 41(10):4554–4562
14. Alligood KT, Sauer TD, Yorke JA (2000) Chaos: an introduction to dynamical systems. Textbooks in mathematical sciences. Springer, Berlin
15. Lorenz EN (1963) Deterministic nonperiodic flow. J Atmos Sci 20:130–148
16. Fraser AM, Swinney HL (1986) Independent coordinates for strange attractors from mutual information. Phys Rev A 33:1134–1140
17. Kennel MB, Brown R, Abarbanel HDI (1992) Determining embedding dimension for phase-space reconstruction using a geometrical construction. Phys Rev A 45(6):3403–3411
18. Narzo A, Narzo F, Aznarte JL, Stigler M (2009) tsDyn: time series analysis based on dynamical systems theory. R package version 0.7
19. Abarbanel HDI, Brown R, Sidorowich JJ, Tsimring LS (1993) The analysis of observed chaotic data in physical systems. Rev Modern Phys 65(4):1331
20. Liebert W, Pawelzik K, Schuster HG (1991) Optimal embeddings of chaotic attractors from topological considerations. Europhys Lett 14(6):521
21. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH (2009) The WEKA data mining software: an update. SIGKDD Explor 11(1)