Eur. Phys. J. Special Topics **223**, 2903–2912 (2014)

DOI: 10.1140/epjst/e2014-02303-y

THE EUROPEAN PHYSICAL JOURNAL SPECIAL TOPICS

Regular Article

Multivariate nonlinear time-series estimation using delay-based reservoir computing

M. Escalona-Morán[1,a], M.C. Soriano[1], J. García-Prieto[2,3], I. Fischer[1], and C.R. Mirasso[1]

[1] Instituto de Física Interdisciplinar y Sistemas Complejos, IFISC (CSIC-UIB), Campus Universitat de les Illes Balears, 07122 Palma de Mallorca, Spain
[2] Centro de Tecnología Biomédica, Universidad Politécnica y Universidad Complutense, 28223 Madrid, Spain
[3] Laboratory of Electrical Engineering and Bioengineering, Dept. of Industrial Engineering, Universidad de La Laguna, Avda. Astrofísico Fco. Sánchez s/n, 38205, La Laguna, Tenerife, Spain

Received 13 July 2014 / Received in final form 17 October 2014
Published online 10 December 2014

Abstract. Multivariate nonlinear time-series analysis represents a major challenge in complex systems science, specially when the full underlying dynamics is unknown. Often, time-series forecast relies on the information contained in a single measured variable. However, in many cases one might benefit from other available measures of the system to improve the prediction of its dynamical evolution. Here, we utilize Reservoir Computing techniques to process sequential multivariate information. As reservoir, we employ a Mackey-Glass delay system. We discuss the approximation of a three-dimensional theoretical model (the Lorenz model) by comparing prediction performance for one variable using either one or two variables as input. Finally, we apply these insights to improve the performance of a relevant biomedical task, namely multi-electrode heartbeat classification.

1 Introduction

Estimation and classification of multivariate time series have become mandatory tasks in many fields of science including neuroscience, genetics, economy, communications technology, social sciences and others. If the generating system is deterministic, it may be possible to reproduce or approximate the dynamics of the system with a constructed model. According to the Takens embedding theorem [1], a single variable of a multivariate time series is sufficient to recover the underlying dynamics, given that the variables are coupled. However, due to noise and other factors, this is limited for real data and time series estimation and classification may benefit from the use of additional measurements [2]. In this paper, we explore the benefits of using multiple data measurements for time series prediction of the Lorenz attractor and electro-cardiogram classification. We present numerical simulations showing that the

[a] e-mail: miguelangel@ifisc.uib-csic.es

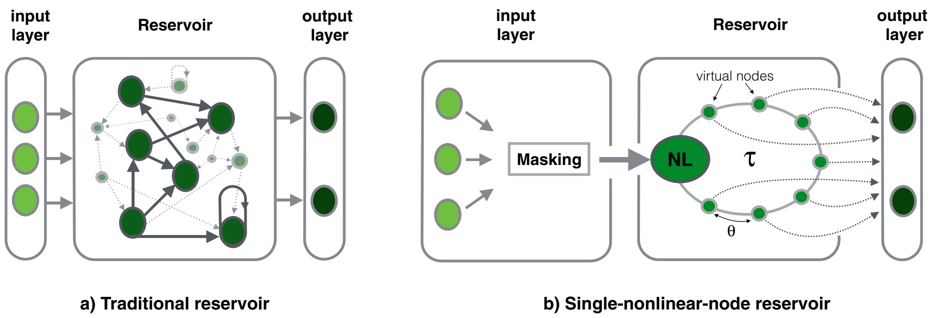

Fig. 1. a) Schematic view of a traditional reservoir computing system. b) Schematic view of reservoir computing based on a single nonlinear node with delay. Virtual nodes are defined as different temporal positions in the delay line. The dashed lines indicate the implicit connection between the virtual nodes and the readout.

use of multiple variables or data measurements can significantly improve prediction and classification.

Among the many different options to estimate time series, machine-learning approaches belong to the most promising ones. Reservoir Computing (RC) (also known as echo state network [3] and liquid state machine [4]) has emerged as one of the most successful machine learning approaches. RC is a neuro-inspired concept that simplifies the traditional Hopfield networks approach, while retaining most of its computational capabilities. The Hopfield networks approach is a well known technique that uses recurrent networks to realize certain computational tasks that are inherently very demanding for traditional computers. In Hopfield networks the network weights connecting the nodes are trained for a specific task and the states of the nodes are readout to process the incoming information. The drawback of this technique is that the complexity of redesigning the reservoir (training the network) grows exponentially whenever the dimensions of the problem grow linearly. This problem is typically referred to as the *curse of dimensionality*. This represents a difficulty since network's weights are difficult to train. The advantage of RC is that the reservoir connections are kept fixed and only the readout weights have to be adapted for each task [5]. This dramatic reduction in the training requirements does not result in a performance degradation. Importantly, it has been shown that RC preserves the properties of recurrent networks [6], namely the capability to compute and to exhibit a fading memory of previous inputs due to the recurrent connections.

Traditional RC systems consist of three layers: an input layer, a reservoir and an output layer (see Fig. 1a). An input signal is injected into the input layer which is often randomly connected to the reservoir. The reservoir is built as a recurrent network whose connectivity has been chosen (often randomly). The states of the network nodes under the presence of the input are read out and linearly combined at the output layer to process the information. The output weights are determined through a training process by using known input signals, similar to those to be tested later. Following this idea, computational demanding tasks like time series prediction or speech recognition [7] as well as the optimization and control of highly complex physical systems [8] have been successfully performed.

In this paper we adopt a modified approach to implement RC. The approach is based on the utilization of a single nonlinear element subject to a delayed-feedback loop [9]. The scheme is shown in Fig. 1b. In this case the nonlinear element is described by the Mackey-Glass equation modified to include an external input $I(t)$

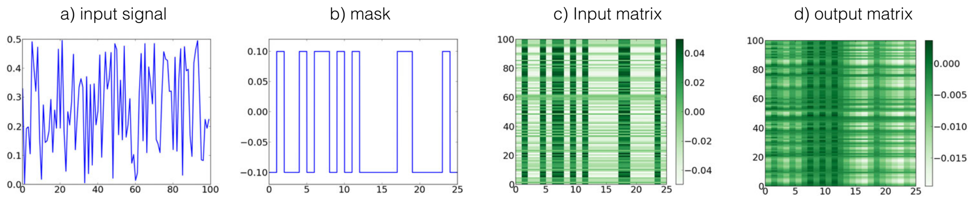

Fig. 2. Schematic view of a traditional reservoir computing system.

and to account for the feedback loop of delay time τ. The equation reads

$$\dot{X}(t) = -X(t) + \frac{\eta \cdot [X(t-\tau) + \gamma \cdot I(t)]}{1 + [X(t-\tau) + \gamma \cdot I(t)]^p}, \tag{1}$$

with X denoting the dynamical variable, $\dot{X}$ its derivative with respect to a dimensionless time t, and τ the delay in the feedback loop. Parameters η and γ represent feedback strength and input scaling, respectively. The exponent p can be used to tune the nonlinearity. For small values of η and without an external input ($\gamma = 0$), the system operates in a stable fixed point. However, in the presence of an external input the system might show complex dynamics. In particular, we are interested in a dynamical regime that produces consistent transient responses, i.e., similar inputs generate similar transient states. Although we have chosen a specific nonlinearity, it has been shown that other nonlinear functions perform equally well [10–13]. The conceptual simplicity of delay-based reservoirs has facilitated the electronic, opto-electronic and all-optical hardware implementations of Reservoir Computing [9,14–18].

As illustrated in Fig. 1b, equally spaced points in the delay line are chosen as *virtual nodes* whose states will be used for the readout process. This follows the insight that the dynamical degrees of freedom for delay systems are distributed along the delay line. The input signal is injected to the nonlinear node with certain input weights following a time-multiplexed masking procedure [9]. In the case of a one-dimensional input signal, each point of the signal to be processed (Fig. 2a) is sampled and held during one delay time τ and multiplied by a binary random mask (Fig. 2b) consisting of $\{-0.1, 0.1\}$, giving rise to a sequence of random input weights. The resulting input to the reservoir is a matrix, $\mathbf{I}$ (Fig. 2c), of dimensions $M \times N$, where M is the number of sampled points in the signal to be processed and N is the number of virtual nodes in the delay line. The input matrix $\mathbf{I}$ is then fed into the nonlinearity in a serial manner, creating a pattern of transient activity in the delay line. The response to every row of $\mathbf{I}$ fills the delay vector of length τ. Once all elements of matrix $\mathbf{I}$ are injected, a state matrix $\mathbf{S}$ (Fig. 2d), which contains the transient responses to the input signal at the virtual nodes, is constructed. These state matrices are used for building an estimator/classifier through a training procedure.

2 Time series estimation: The Lorenz system

The prediction of chaotic time series is a demanding task due to the sensitive dependence on initial conditions and the intricate geometric structures of the corresponding chaotic attractors. In the following example, we consider time series generated by the Lorenz system [19] which is described as

$$\begin{aligned} \dot{x} &= \sigma(y - x), \\ \dot{y} &= x(\rho - z) - y, \\ \dot{z} &= xy - \beta z. \end{aligned} \tag{2}$$

In our particular case, we choose parameters as follows: $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$. In order to perform the time series estimation, different numerical realizations of the Lorenz system are computed using random initial conditions and an integration time step of $\Delta t = 0.01$. Then the time series are downsampled by a factor of 20, resulting in a final sampling step of 0.2. For our analysis, each time series has a final length of 10^3 points. It is worth mentioning that results shown in this article are for the above mentioned length of the time series.

To build the estimator, 24 time series, generated from Eq. (3) using different initial conditions, are used for training and one for testing. Then the procedure is repeated until all time-series samples are used for training and testing. This procedure is known as a 25-random-fold cross-validation evaluation [20,21].

As it is typical in RC, a linear regression method is used to computed the weights of the virtual nodes. For this task, 400 virtual nodes are used to capture the dynamics of the system. In the case of time series prediction using a single variable as the input to the reservoir, the mask contains two values $\{-0.01, 0.01\}$ assigned randomly to the corresponding virtual nodes. In the case of time series prediction using two variables of the Lorenz system as input to the reservoir, the mask then contains three different values $\{-0.01, 0.0, 0.01\}$ assigned randomly following a proportion of $[30, 40, 30]\%$ respectively. Note that the mask is a matrix of dimensions $2 \times N$ in the case of 2-dimensional input signals.

We choose the Mackey-Glass equation (Eq. (1)) as the reservoir. The exponent $p = 1$ allows for a long fading memory in the reservoir, which is important in the context of time series prediction.

To solve our task, the parameter η in Eq. (1) is chosen as $\eta = 0.45$, after exploring the parameter space $\eta - \gamma$ and finding a good performance for this parameter value. To measure the accuracy of our prediction, we compute the Normalized Mean Square Error (NMSE) which is an estimator of the overall deviations between predicted and measured values. The NMSE is defined as,

$$NMSE = \frac{1}{m} \frac{\sum_{k=1}^{m} (target_k - input_k)^2}{\sigma^2(t_k)}, \tag{3}$$

were m is the number of samples in the time series: *input* represents the original input signal and *target* is the predicted time series. σ denotes the standard deviation (STD).

Time series prediction is often performed using a single variable since other variables are usually hidden or not accessible. However, we will show below that if one has access to other variables of the system, the prediction can be significantly improved. For the Lorenz system we concentrate on the variables x and y since they are the most difficult to predict because they move between the two wings of the Lorenz strange attractor. We initially predict the values of the x one step ahead in time (a test called one-step prediction task) in two ways. First, based only on its own trace (but only a single point of x is presented to the reservoir at each delay period) or on the time series of the y variable. Second, the prediction is based on its own time series and the time series of the y variable, i.e. we perform a multivariate prediction of the x variable. In Fig. 3 we plot the NMSE of the one-step prediction as a function of the input scaling γ for the variable x using its own trace (red lines and crosses) and using only the trace of y (green line and plus signs). It can be seen that a better prediction is obtained when using the own time series of the x variable, with a NMSE that does not depend much on $\gamma(<1)$. This highlights the robustness of the results to changes in parameter values. For values of $\gamma > 1$ the prediction degrades since we place the system beyond the bifurcation point of the Mackey-Glass equation which affects the consistency of the responses.

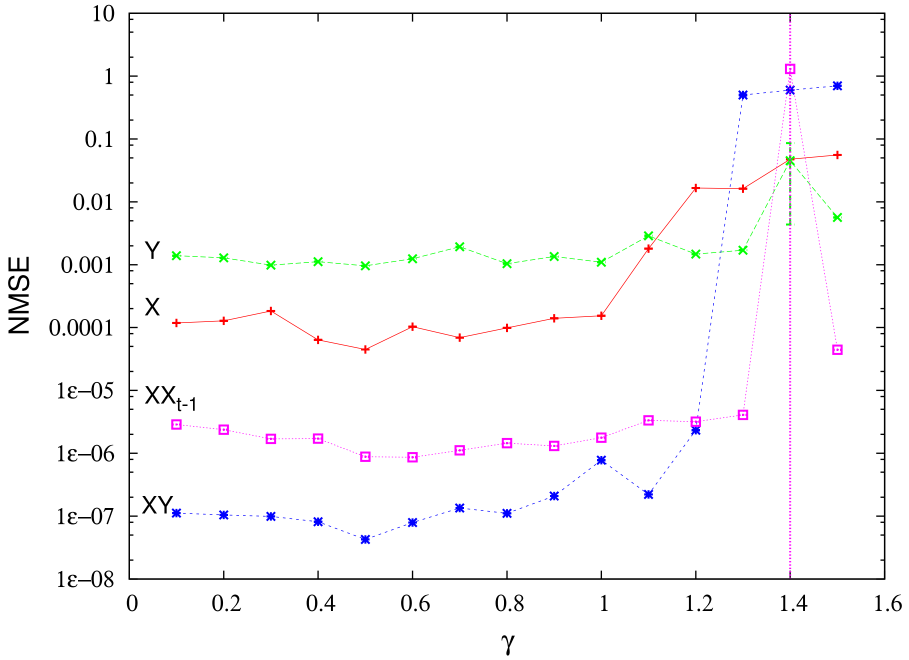

Fig. 3. NMSE as a function of γ for the estimation of variable x. Green curve (crosses) corresponds to the NMSE of x when using only y variable. Red curve (plus signs) corresponds to the NMSE of estimating x when using only x variable. Blue curve (stars) corresponds to the NMSE of x when using both x and y variables. Magenta curve (squares) is the error of estimating x when using x and its delayed version, x_{t-1}. Error bars were computed as the STD of the 25 different folds.

To check whether the prediction can be improved, by taking into account its own time series and that of the variable y, we again compute for this case the NMSE of the x variable. As shown in Fig. 3, the NMSE significantly reduces, reaching values around 10^{-7} for $\gamma < 1$. When comparing the results for x using x trace and using x and y traces, it can be seen that we have reduced the NMSE by about three orders of magnitude. It is worth noting that the prediction can also be improved if a delayed version of the x variable is given as an input to the reservoir. However, in this case, including x and y variables is more significant than including the x variable and its delayed version (magenta curve and square signs).

To substantiate our results further, we compute the prediction error of x for more than one time step ahead. In Fig. 4 it can be seen that the NMSE remains below 1% up to 11 predicted points ahead when using both x and y variable time series. When using only the time series of x only 7 points ahead can be predicted withing the same error bounds. Therefore, the prediction of x is clearly improved when both x and y are presented as input to the reservoir.

Time series prediction is one of the most common tasks in machine learning as well as classification tasks. In the next section we present a real-world classification problem that can help for the diagnosis and exploration of cardiovascular diseases. We will apply a similar multivariate analysis to study the classification of ECG signals when a single or two channels are taken into account to classify healthy and pathological subjects with a delay-based reservoir.

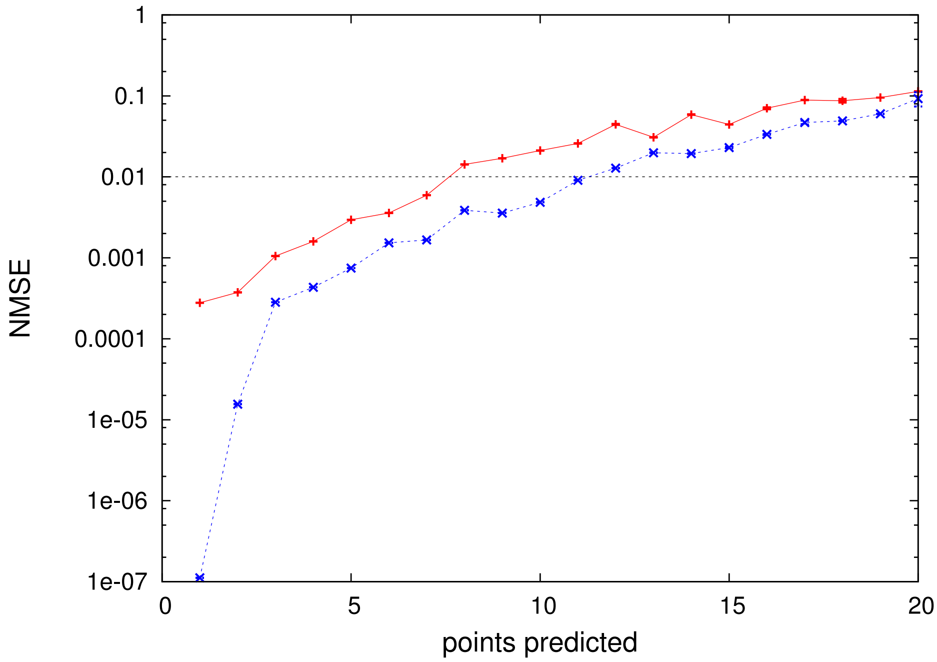

Fig. 4. NMSE as a function of the predicted steps for the variable x. Red curve (plus signs) corresponds to the NMSE of x when using only its own time series. Blue curve (stars) corresponds to the NMSE of x when using both x and y time series. Error bars were computed as the STD of the 25 different folds.

3 Electrocardiographic signal classification

Cardiovascular disease (CVD) is the primary cause of death throughout the world, accounting for 24% of all deaths globally [22]. Electrocardiograms (ECG) have been a powerful and irreplaceable tool in the exploration and diagnostic of CVDs. Its acquisition requires only simple and low-cost devices with a minimum impact on the patient. The ECG is an essential diagnostic tool for common pathologies such as myocardial ischemia [23,24], arrhythmia and other rare pathologies as cardiac muscular dystrophy or Brugada syndrome [25]. Continuous monitoring in medicine has given physicians the ability to collect hours and days worth recordings of physiological signals, involving physicians in a time-consuming analyzing process. This has prompted researchers to develop automatic diagnosis tools for the detection of cardiac diseases in order to reduce the diagnosis time.

In the following, we choose the MIT-BIH Arrhythmia Database [26] which contains 48 ambulatory ECG recordings of half hour each, in order to test our methodology with this real-world problem. Twenty-three recordings were selected at random while the remaining 25 recordings were selected to include less common but clinically significant arrhythmias. Each ECG record includes two time series originating from different electrodes. The most common derivation in the database is the modified limb lead II (MLII), present in 41 recordings. The second most common derivation is $V1$ present in 35 recordings. We have restricted our database to 35 recordings in order to include these two derivations for our multivariate analysis.

To perform heartbeat classification, the ECGs were divided into heartbeats using a fixed-length window of 170 samples around the R-peak. This particular point of the ECG is annotated in the database. The window was positioned around the maximum

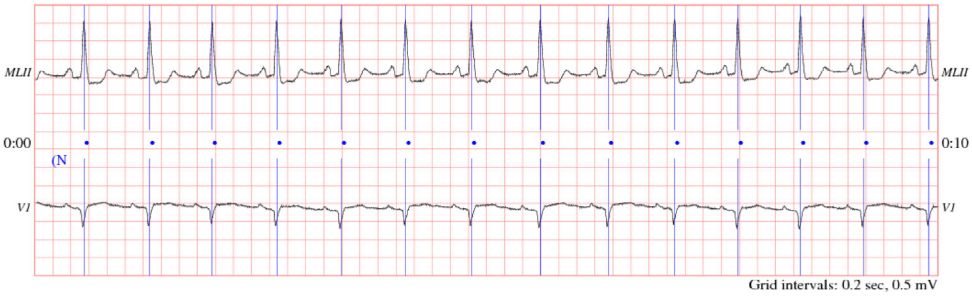

Fig. 5. ECG traces of the two main derivations, namely MLII and V1, for a normal subject. Vertical lines represent the position of the R-peak. Figure taken from Physionet.org

peak of the QRS complex to extract the waveform (see Ref. [28] for details). Seventy samples before the R-peak were extracted to include P waves, and 100 samples after the R-peak were also included to have information about the T wave and the duration of the heartbeat. Figure 5 shows the beginning of an ECG of a normal subject. Both, MLII and V1 channels are represented. Vertical lines represent the position of the R-peak. The N in the plot means that the heartbeats are normal.

For this particular application, we employ logistic regression for the training procedure. This is in contrast to standard Reservoir Computing which uses linear regression methods. The logistic regression (LR) [27] is a widely used learning technique in biostatistical applications in which binary responses occur quite frequently, i.e. for questions whether a condition is present or absent. LR is specified in terms of logit transformations, defined as

$$\mathrm{logit}(P) = \ln(\mathrm{odds}) = \ln\left(\frac{P}{1-P}\right) \tag{4}$$

where the odds represent the ratio of the probability P that an event will occur to the probability that the same event will not occur. In the logistic regression, the aim is to linearly relate the logit function with the data $\mathbf{S}$ finding the values of parameters a and b that satisfy

$$\mathrm{logit}(\mathbf{P}) = a + b \cdot \mathbf{S}, \tag{5}$$

where $\mathbf{S}$ are the state matrices and each component of $\mathbf{P}$ contains the probability for the corresponding state matrix. Consequently, results can be directly interpreted as the probability of a condition to be true or false via the following equation

$$\mathbf{P} = \frac{e^{a+b\mathbf{S}}}{1+e^{a+b\mathbf{S}}} \,. \tag{6}$$

Note that logit functions are linearly related to the data $\mathbf{S}$, but the probabilities are nonlinearly related to it. This is an advantage since in classical linear models it is usually assumed that the outcomes are independent and normally distributed with equal variance. These assumptions are often inadequate in medical applications.

For the classification of the ECG signals, we again employ the Mackey-Glass delay system as the reservoir. We have numerically checked that, for this task, the optimal number of virtual nodes is 25. A smaller number of nodes leads to a high bias problem, i.e. the reservoir is too simple to classify the different ECG signals. A larger number of nodes (>25) leads to a high variance problem (overfitting), i.e. the reservoir is able to learn the training samples but it is not able to generalize to new samples. The optimal number of nodes can be extracted from the learning curves of the system [28]. In [28], we have also explored the parameter space of the Mackey-Glass model

Table 1. Accuracy of the classification of ECGs using one or two channels.

Channel	Accuracy (%)
MLII	71.5
MLII/V1	78.6

finding a richer dynamics on the state matrices for $\eta = 0.8$ and $\gamma = 0.5$. In addition, we choose parameter $p = 7$ in Eq. (1). For this value the Mackey-Glass oscillator exhibits shorter memory but a higher nonlinearity, which is more convenient for a classification task.

One standard measure for the performance of a classifier is the accuracy measure, which is defined as:

$$\text{Accuracy} = \frac{\text{true positive} + \text{true negative}}{\text{sum of all samples}}\,. \tag{7}$$

For further information on the evaluation of performance in machine learning, please refer to [29].

For a clinically-relevant, multiclass, ECG classification problem of 45 subjects from the MIT-BIH Arrhythmia Database the average accuracy of the classifier was 98.43% [28], showing improvement over previously reported results.

Here, and in contrast to [28], we construct the classifier from two channels, namely the MLII and the V1 channels. We have also increased the tolerance of the algorithm to 10^{-1} to leave room for improvement on the results shown in [28]. These two channels are only available for a reduced set of patients (35 subjects), leading to a smaller usable database and a degradation in performance. Table 1 shows the accuracy of the classifier when using one or two channels, for the restricted database containing both derivations. The classification for a single channel is performed using channel MLII alone. Then a combination of channels MLII and V1 is used to build a multivariate classifier. It can be observed that the accuracy increases about 7% when using two variables in comparison with the case of one variable. This is in agreement with results reported by De Chazal [30] and highlight the potential of multivariate timeseries prediction using reservoir computing. More tests and larger data sets are, however, needed to further verify the usefulness of our approach.

4 Conclusions

In this paper we have numerically shown the ability of reservoir computing, based on delay-coupled systems, to perform time series prediction and classification tasks following a multivariate analysis. We have concentrated on two tasks, the prediction of a chaotic time series, given by the Lorenz system, and the classification of heart beats, obtained from ECG derivations.

For the one-step prediction task of the Lorenz system, we found a significant reduction (∼3 orders of magnitude) of the normalized mean square error (NMSE) when using two variables to predict one, than when using only one variable. Moreover, we found that the NMSE remains smaller than 1% when predicting 11 steps ahead when using two variables compared to only 7 steps using one variable. We expect the results obtained for the well-known Lorenz system in the chaotic regime to be valid for similar multivariate dynamical systems.

We have also applied the multivariate approach to the classification of heart beats. We found an improvement of 7% when using two channels of the ECG as compared to the case when the classification was performed using a single channel. In this context, the accuracy in the classification can improve if a database with a larger number of subjects is used.

Our results highlight that the use of more than one variable can significantly improve predictions when using reservoir computing techniques. More tests with real-world data are however needed to explore the full potential of our approach.

It is worth noting that the fading memory present in recurrent networks resembles the time-delay embedding in Takens theorem. This fading memory implies that information about previous inputs is still present in the reservoir after a number of delay times τ. In our case, adding explicitly a delayed version of the same input does not provide as much information as adding the y variable. Thus, we find a significant improvement in the prediction capabilities of reservoir computing when using an additional variable.

References

1. F. Takens, in *Dynamical systems and turbulence, Warwick 1980* (Springer, 1981), p. 366
2. L. Cao, A. Mees, K. Judd, Physica D: Nonlinear Phenom. **121**(1), 75 (1998)
3. H. Jaeger, Bonn, Germany: German National Research Center for Information Technology GMD Technical Report **148** (2001)
4. W. Maass, T. Natschläger, H. Markram, Neural Comput. **14**(11), 2531 (2002)
5. K. Vandoorne, W. Dierckx, B. Schrauwen, D. Verstraeten, R. Baets, P. Bienstman, J. Van Campenhout, Opt. Express **16**(15), 11182 (2008)
6. M. Lukoševičius, H. Jaeger, Comput. Sci. Rev. **3**(3), 127 (2009)
7. D. Verstraeten, B. Schrauwen, M. d'Haene, D. Stroobandt, Neural Netw. **20**(3), 391 (2007)
8. M. Hermans, B. Schrauwen, P. Bienstman, J. Dambre, PLoS ONE **9**(1), e86696 (2014)
9. L. Appeltant, M.C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C.R. Mirasso, I. Fischer, Nature Commun. **2**, 468 (2011)
10. J. Dambre, D. Verstraeten, B. Schrauwen, S. Massar, Sci. Rep. **2** (2012)
11. K. Hicke, M. Escalona-Morán, D. Brunner, M.C. Soriano, I. Fischer, C. Mirasso, IEEE J. Selected Topics in Quantum Electron. **19**(4), 1501610 (2013)
12. M.C. Soriano, S. Ortín, D. Brunner, L. Larger, C.R. Mirasso, I. Fischer, L. Pesquera, Opt. Express **21**(1), 12 (2013)
13. R.M. Nguimdo, G. Verschaffelt, J. Danckaert, G. Van der Sande, Opt. Express **22**(7), 8672 (2014)
14. R. Martinenghi, S. Rybalko, M. Jacquot, Y.K. Chembo, L. Larger, Phys. Rev. Lett. **108**(24), 244101 (2012)
15. L. Larger, M.C. Soriano, D. Brunner, L. Appeltant, J.M. Gutiérrez, L. Pesquera, C.R. Mirasso, I. Fischer, Opt. Express **20**(3), 3241 (2012)
16. Y. Paquot, F. Duport, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman, S. Massar, Sci. Rep. **2** (2012)
17. F. Duport, B. Schneider, A. Smerieri, M. Haelterman, S. Massar, Opt. Express **20**(20), 22783 (2012)
18. D. Brunner, M.C. Soriano, C.R. Mirasso, I. Fischer, Nature Commun. **4**, 1364 (2013)
19. E.N. Lorenz, J. Atmos. Sci. **20**(2), 130 (1963)
20. P.A. Devijver, J. Kittler, *Pattern recognition: A statistical approach*, Vol. 761 (Prentice-Hall London, 1982)
21. R. Kohavi, et al., *A study of cross-validation and bootstrap for accuracy estimation and model selection*, in *IJCAI*, Vol. 14 (1995), p. 1137
22. M. Heron, National Vital Statistcs Report **62**(6) (2013)
23. H.P. Selker, R.J. Zalenski, E.M. Antman, T.P. Aufderheide, S.A. Bernard, R.O. Bonow, W.B. Gibler, M.D. Hagen, P. Johnson, J. Lau, et al., Ann. Emergency Med. **29**(1), 1 (1997)

24. J.H. Pope, H.P. Selker, Emergency Med. Clinics North Amer. **21**(1), 27 (2003)
25. P. Brugada, J. Brugada, J. Am. College Cardiology **20**(6), 1391 (1992)
26. R. Mark, P. Schluter, G. Moody, P. Devlin, D. Chernoff, Front. Eng. Health Care (1982), p. 205
27. Y. Pawitan, *In all likelihood: statistical modelling and inference using likelihood* (Oxford University Press, 2001)
28. M. Escalona-Morán, M.C. Soriano, I. Fischer, C.R. Mirasso, IEEE J. Biomedical Health Informatics, doi: 10.1109-JBHI.2014.2332001 (2014)
29. D.M.W. Powers, J. Mach. Learning Technol. **2**(1), 37 (2011)
30. P. De Chazal, M. O'Dwyer, R.B. Reilly, Biomed. Eng., IEEE Trans. **51**(7), 1196 (2004)