

Analysis of the Dynamics of the Echo State Network model using Recurrence Plot

Emmanuel Sam¹, Sebastian Basterrech², and Pavel Kromer³

¹ Nduom School of Business and Technology, Elmina, Central Region, Ghana
emsam@nsbt.edu.gh

² Department of Computer Science, Faculty of Electrical Engineering
Czech Technical University, Prague, Czech Republic.
Sebastian.Basterrech@fel.cvut.cz

³ Faculty of Electrical Engineering and Computer Science
VŠB-Technical University of Ostrava, Ostrava, Czech Republic
Pavel.Kromer@vsb.cz

Abstract. At the beginning of the 2000s, a specific type of Recurrent Neural Networks (RNNs) was developed with the name Echo State Network (ESN). The model has become popular during the last 15 years in the area of temporal learning. The model has a RNN (named reservoir) that projects an input sequence in a feature map. The reservoir has two main parameters that impact the accuracy of the model: the reservoir size (number of neurons in the RNN) and the spectral radius of the hidden-hidden recurrent weight matrix. In this article, we analyze the impact of these parameters using the Recurrence Plot technique, which is a useful tool for visualizing chaotic systems. Experiments carried out with three well-known dynamical systems show the relevance of the spectral radius in the reservoir projections.

Keywords: Recurrent Neural Network, Echo State Network, Recurrence Plot, Chaotic Systems, Time-series Problems

1 Introduction

Recurrent Neural Networks (RNNs) are neural networks with cyclic path of connections among their neurons [1]. Due to this underlying property, they possess powerful computational and dynamical memory capabilities which make them suitable for modeling nonlinear relationships among sequential and temporal data. In the early 2000s, Echo State Network (ESN) [2] and a closely related approach known as Liquid State Machine (LSM) [3], introduced a new computational framework for training RNN. This framework, which has lately become known as Reservoir computing (RC) [4], demonstrates that RNN can still perform significantly well even when only a subset of the network weights are trained. In this approach, a randomly initialized RNN, known as reservoir, improves the linear separability of the input data. The reservoir (a matrix with the hidden-hidden weights) projects the input data in a feature space, then a supervised model is used to perform the outputs. Due to its simplicity, robustness,

computational speed, and ease of implementation, RC has become popular in the Artificial Neural Network Community [5]. It has yielded successful results in many benchmark problems [5]. The ESN model and its variations have also been successfully applied on practical problems such as time series predictions [6, 5] and pattern classification [2].

The computational power of ESN is based largely on the reservoir structure, and therefore the design of the reservoir and its characteristics have been the focus of many RC research over the years [7–9]. The standard ESN reservoir is influenced by a number of global parameters, which impact in the model accuracy. The most relevant ones are: sparsity and spectral radius of the reservoir matrix and dimension of the reservoir matrix [10]. The spectral radius of the reservoir matrix is related to a fundamental algebraic property, known as Echo State Property (ESN) [2], that ensures the state of the reservoir is suitable for good predictions. Guidelines on how the spectral radius can be tuned to guarantee good performance of ESN can be seen in [2]. In this study, we exploit the power of Recurrence Plot (RP) [11], a visual representation of the recurrences of dynamical systems, to investigate the effect of a given reservoir size and spectral radius combination on the dynamics of ESN reservoir. We generate several reservoir architectures with a given set of parameters and feed it with a benchmark signal. Then, we apply RP to analyze the reservoir projections. We experiment with three well-known benchmark datasets which include Henon, Lorenz, and Rossler dataset. A description of these benchmarks can be found in [7]. The possibility of exploring the dynamics of ESN and analyzing the stability of the recurrences has also been studied in [12]. The authors analyzed the effect of the input signal in the dynamics using RP and Recurrence Quantification Analysis (RQA) over two signals: sinusoidal waveform and Mackey-Glass time-series. In this article, we focus on analyzing the impact of the pair: reservoir size and spectral radius in the stability and accuracy of the ESN model using RP.

The rest of this paper is organized as follows. Section 2 describes the ESN model and its properties, and reviews relevant literature on ESN. Section 3 provides a description of the methodology for this study. Experimental results and their related explanations are presented in Section 4. We end with a discussion and recommendations for future work.

2 Description of Echo State Network

An ESN is made up of two main distinct structures: a random initialized and fixed hidden-hidden weights matrix called reservoir and a parametric mapping often a linear regression called readout. When its input neurons are driven by a signal, $\mathbf{s}(t)$ at any time t , the reservoir acts as a dynamical system that transforms the original input signal from an input space \mathbb{R}^p into a larger space \mathbb{R}^d with $p \ll d$, using a high dimensional feature map. Like kernel functions, this enhances the linear separability of the input data. Additionally, the recurrent matrix memorizes the sequence of input patterns, making the ESN suitable for solving temporal learning problems. The readout structure is a parametric map-

ping (often linear) from the feature map created by the reservoir and the output space. A characteristic of the model is that it does not train the hidden-hidden weights; only the readout parameters are trained. As a consequence, the model is fast and robust, and the problem of vanishing-exploding gradient [13] that is often presented in the training of RNNs is avoided. Several extensions and variations of the standard ESN model have been proposed in the literature. Examples include: intrinsic plasticity [14], BackPropagation-Decorrelation [15], Decoupled ESN [16], Leaky integrator [17], Evolino [18], and a recently introduced Echo State Network based on Queuing Theory [19, 6].

In this study we consider an ESN with standard topology containing an input layer with p neurons connected to d hidden neurons, and a readout layer with o neurons. We assume that both input and output signals are real values, and we consider discrete time. The input weight matrix for the connections between the input neurons and reservoir neurons is denoted by \mathbf{W}^{in} whilst the weight matrix for the internal connections inside the reservoir and the weight matrix for connections between the reservoir and readout layer are denoted by \mathbf{W}^{r} and \mathbf{W}^{out} respectively. The dimensions of these matrices are $d \times (1 + p)$, $d \times d$, and $o \times (1 + p + d)$, respectively. The first row of \mathbf{W}^{in} and \mathbf{W}^{out} contains a value corresponding to bias terms.

Given a training set composed of input signal $\mathbf{s}(t) \in \mathbb{R}^p$ the reservoir updates its activation state $\mathbf{x}(t) = (x_1(t), \dots, x_N(t))$ using an activation function $g_h(\cdot)$ with parameters \mathbf{W}^{in} and \mathbf{W}^{r} as follows:

$$\mathbf{x}(t) = g_h(\mathbf{s}(t), \mathbf{x}(t-1), \mathbf{W}^{\text{in}}, \mathbf{W}^{\text{r}}). \quad (1)$$

Next, the parametric function shown below uses the actual reservoir states to execute the model output:

$$\hat{\mathbf{y}}(t) = g_o(\mathbf{x}(t), \mathbf{W}^{\text{out}}),$$

where $g_h(\cdot)$ is an activation function with parameters in \mathbf{W}^{out} . Although in the standard ESN model there are no connections between the input and readouts neurons [3, 20], another readout form is the following:

$$\hat{\mathbf{y}}(t) = g_o(\mathbf{s}(t), \mathbf{x}(t), \mathbf{W}^{\text{out}}). \quad (2)$$

In this study, we used hyperbolic tangent $\tanh(\cdot)$ as the activation function $g_h(\cdot)$, and the dynamics was computed as:

$$\mathbf{x}(t) = \tanh(\mathbf{W}^{\text{in}}\mathbf{s}(t) + \mathbf{W}^{\text{r}}\mathbf{x}(t-1)). \quad (3)$$

The output of the model, $\hat{\mathbf{y}}(t)$ at a given time t is computed as follows:

$$\hat{\mathbf{y}}(t) = \mathbf{W}^{\text{out}}[\mathbf{s}(t); \mathbf{x}(t)], \quad (4)$$

where $[\cdot; \cdot]$ denotes a vector concatenation operation.

3 Methodology

We conducted experiments on three popular signals: Henon, Rossler, and Lorenz datasets. A detailed description of the datasets can be found in [7]. The samples in the dataset were initially normalized to lie between 0 and 1, and the resulting data was divided into two subsets: 80% was used to train the ESN and the remaining 20% was used to test it. As mentioned above, we consider a standard ESN where \mathbf{W}^{in} and \mathbf{W}^{r} are randomly initialized with uniformly distributed weights in the range $[-0.5, 0.5]$. For each dataset, the reservoir was configured with different combinations of reservoir size d and spectral radius ρ . We consider the pair (d, ρ) with values in the grid generated by $d = \{100, 250, 500\}$ and $\rho = \{0.1, 0.5, 0.99\}$. The reservoir states are updated with a leaking rate of 0.3 and the output weights are estimated by setting the regularization factor in the linear regression model to 1×10^{-3} . The trained ESN was ran in a generative mode (i.e. previous predictions were fed back into the reservoir as input for the next prediction), and the effect of a selected pair of parameters (d, ρ) , on the accuracy of the ESN model was measured with Mean Square Error (MSE). The corresponding dynamical properties of the ESN reservoir was visualized using Recurrence Plot (RP) [11]. In the following we present a brief introduction of the RP technique.

We use a subset X_{sub} that collects n states during the training of the ESN model to generate the recurrence matrix R , to stand for the recurrences of the reservoir states. In the experimental results we set up the same value of $n = 500$ for all the signals. Given any multidimensional signal $\mathbf{x}(i)_{i=1}^n$, then the corresponding RP is based on the following matrix:

$$R_{i,j} = \begin{cases} 1 : \mathbf{x}(i) \approx \mathbf{x}(j), & i, j = 1, \dots, n, \\ 0 : \mathbf{x}(i) \not\approx \mathbf{x}(j), & \end{cases}$$

where n is the number of states considered. We say that $\mathbf{x}(i) \approx \mathbf{x}(j)$ if an arbitrary distance function is lower than an arbitrary value ϵ . In our case, we consider the sequence of reservoir states. Therefore we created the binary entries of R using the following rule [11]:

$$R_{i,j} = \begin{cases} 1, & \text{if } \|\mathbf{x}(i) - \mathbf{x}(j)\| < \epsilon, \quad i, j = 1, \dots, n, \\ 0, & \text{otherwise,} \end{cases} \quad (5)$$

where $\|\cdot\|$ is the L_2 - norm (Euclidean norm), and ϵ is a threshold distance, computed using a percentage of the distance between the maximum L_2 - norm and the minimum L_2 - norm of X_{sub} . There are several variations of the RP technique, the main differences among them are the type of distance functions and epsilon values [11]. In the experimental results we visualize the reservoir matrix using three threshold distances: ϵ_1 , ϵ_2 , and ϵ_3 , computed using the 10%, 50%, and 100% of the distance between the maximum and the minimum values in the L_2 - norm of X_{sub} .

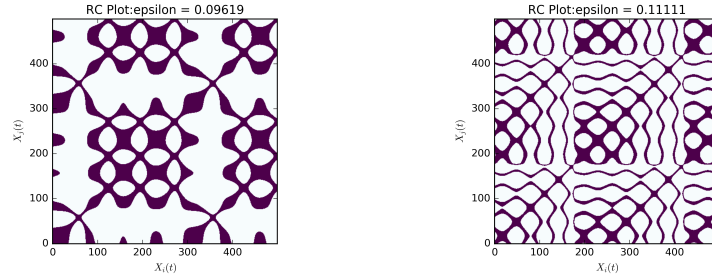
4 Experimental results

In this section we present experimental results related to different configurations of the ESN reservoir. For each benchmark signal, we discuss the effect of each combination of reservoir size and spectral radius (d, ρ) on the accuracy of ESN, and interpret the dynamics of the reservoir using their associated RPs. Table 1 presents the accuracy for each of the analyzed dataset. The table shows the MSE according to the pairs (d, ρ) . In the case of an ESN with a reservoir of 500 neurones and a spectral radius of 0.1 we obtained unstable results for the Rossler dataset, therefore it is not presented in the table.

Figures 1a and Figure 1b present RPs representing the input signals from the Lorenz and Rossler data sets respectively. Figures 2a and 2b show the RPs and MSEs related to two different values of ρ with 100 neurons when driven by input signals from the Henon dataset. Though the recurrence matrices were obtained with similar ϵ (i.e. $\epsilon = 0.281$ in Figure 2a and $\epsilon = 0.282$ in Figure 2b), the MSEs are different and the RP in Figure 2b is more sparse. Figures 3a and 3b present results based on the Lorenz dataset. The figures represent the RPs obtained with $\epsilon = 0.45$ (in Figure 3a) and 0.55 (in Figure 3b), for a reservoir with 100 neurons and spectral radius of 0.1 (in Figure 3a) and 0.99 (in Figure 3b). Note that, the RP shown in Figure 3b was obtained with an ϵ value larger than the one shown in Figure 3a. In spite of this, the RP created with $\rho = 0.1$ is much more dense than the RP visualization created with an ESN with spectral radius $\rho = 0.99$. Besides, the MSE for Figure 3b is lower than that for Figure 3a. Figures 4a and 4b represent the projections of the reservoir when the patterns are from the Rossler dataset. The reservoir has 100 neurons and the ϵ has values 0.47056 (in Figure 4a) and 0.59753 (in Figure 4b) and spectral radius of 0.1 (in Figure 4a) and 0.99 (in Figure 4b). Similar to the case of Lorenz dataset, the reservoir configuration which reaches lower MSE is the one that is more sparse when visualized using RP (i.e. Figure 4b). Even though the ϵ of Figure 4a is lower than ϵ that of Figure 4b, the visualization presented in Figure 4a is much more dense than the one presented in Figure 4b. Figures 5a and 5b show visualizations of the reservoir projection for the Lorenz problem. Both figures have same spectral radius but different reservoir sizes. As a consequence, we can see how the reservoir size does not present a relevant impact in the recurrence dynamics.

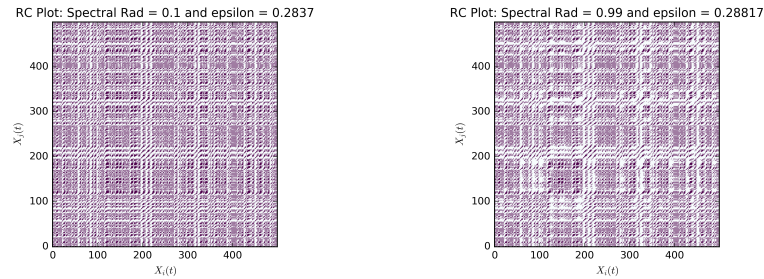
Table 1: ESN accuracy using different network architecture.

Data	d	ρ	MSE	Data	d	ρ	MSE	Data	d	ρ	MSE
Henon	100	0.1	0.16013185	Lorenz	100	0.1	0.05212761	Rossler	100	0.1	0.06859436
	100	0.5	0.15717055		100	0.5	0.12274431		100	0.5	0.03451661
	100	0.99	0.14910897		100	0.99	0.09936286		100	0.99	0.02423478
	250	0.1	0.16173990		250	0.1	0.12022802		250	0.1	-
	250	0.5	0.16061450		250	0.5	0.08143574		250	0.5	0.06668869
	250	0.99	0.15219079		250	0.99	0.11635624		250	0.99	0.01436883
	500	0.1	0.16722330		500	0.1	0.13998824		500	0.1	-
	500	0.5	0.14815264		500	0.5	0.08130847		500	0.5	0.07337573
	500	0.99	0.15553267		500	0.99	0.07223923		500	0.99	0.00776130



(a) Visualization using RP of a time-windows of the Lorenz dataset. (b) Visualization using RP of a time-windows of the Rossler dataset.

Fig. 1: Visualization of the original sequential data.



(a) MSE: 0.15057836.

(b) MSE: 0.16286800.

Fig. 2: Henon dataset: RPs created with 100 reservoir neurons and a spectral radius of 0.1 (Figure 2a) and 0.99 (Figure 2b), and epsilon of 0.281 (Figure 2a) and 0.282 (Figure 2b).

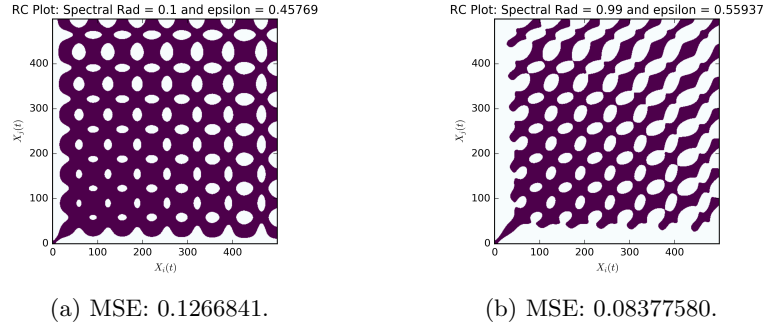


Fig. 3: Lorenz dataset: RPs created with 100 reservoir neurons and a spectral radius of 0.1 (left figure) and 0.99 (right figure), and epsilon of 0.45 (left figure) and 0.55 (right figure).

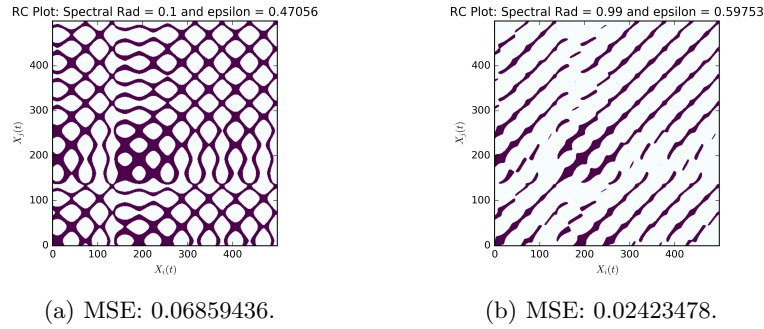


Fig. 4: Rossler dataset: RPs created with 100 reservoir neurons and a spectral radius of 0.1 (Figure 4a) and 0.99 (Figure 4b), and epsilon of 0.47056 (Figure 4a) and 0.59753 (Figure 4b).

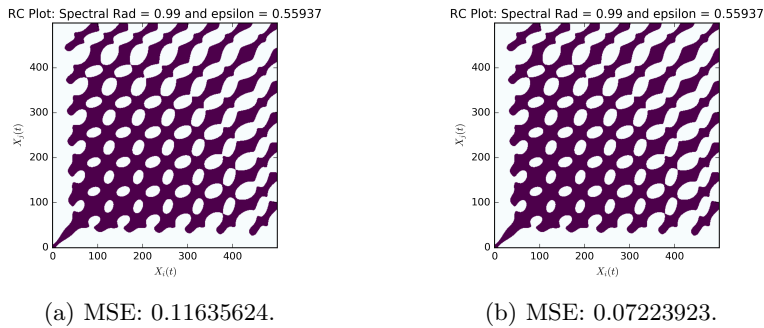


Fig. 5: Lorenz dataset: RPs created with matrices with same spectral radius ($\rho = 0.99$), and different reservoir size. Figure 5a was made with a reservoir with 250 neurons and Figure 5b was made with a reservoir of 500 neurons.

5 Conclusions and Future Work

In this paper, we have investigated the effect in the accuracy of the two main parameters of the Echo State Network (ESN) model. We used Recurrence Plots (RPs) for visualizing the recurrences generated by the phase space of the projections built by the hidden-hidden weight matrix (reservoir). We can infer from the experiments that, although both the reservoir size and spectral radius have significant effect on the accuracy of the model, the spectral radius of the reservoir matrix is much more relevant than the reservoir size as far as the projections are concerned. In other words, the sequence of reservoir states present similar characteristics, regardless of the reservoir size. However, the sequence of reservoir state is characterized by the spectral radius of the reservoir matrix. We noticed that, regardless of the epsilon used to obtain the RP, lower spectral radius (i.e. $\rho = 0.1$) lead to dense RP as well as higher MSE, and higher spectral radius ($\rho = 0.99$) lead to sparse RP as well lower MSE in most cases. Thus, another relevant result is that we found a relationship between the model accuracy and the RP matrix. In general, we obtain better results when the RP matrix is sparse and it has a form similar to the original input signal. In future work, we would like to analyze the relationship between the RP matrix and the memory capacity of the model.

Acknowledgment

This work was supported by the Czech Science Foundation under the grant no. GJ16-25694Y, and by the projects SP2018/126 and SP2018/130 of the Student Grant System, VSB-Technical University of Ostrava, and it has been supported by the Czech Science Foundation (GAČR) under research project No. 18-18858S.

References

1. J. Schmidhuber, "Deep learning in neural networks: An overview." *Neural Networks*, vol. 61, pp. 85–117, 2015.
2. H. Jaeger, "The "echo state" approach to analysing and training recurrent neural networks," German National Research Center for Information Technology, Tech. Rep. 148, 2001.
3. W. Maass, T. Natschläger, and H. Markram, "Real-time computing without stable states: a new framework for a neural computation based on perturbations," *Neural Computation*, vol. 14, pp. 2531–2560, november 2002.
4. D. Verstraeten, B. Schrauwen, M. D'Haene, and D. Stroobandt, "An experimental unification of reservoir computing methods," *Neural Networks*, vol. 20, no. 3, pp. 287–289, 2007.
5. M. Lukoševičius and H. Jaeger, "Reservoir Computing Approaches to Recurrent Neural Network Training," *Computer Science Review*, vol. 3, pp. 127–149, 2009.
6. S. Basterrech and G. Rubino, "Echo State Queueing Networks: A Combination of Reservoir Computing and Random Neural Networks," *Probability in the Engineering and Informational Sciences*, vol. 31, pp. 457–476, October 2017. [Online]. Available: <https://doi.org/10.1017/S0269964817000110>

7. S. Basterrech, "Empirical analysis of the necessary and sufficient conditions of the echo state property," in *2017 International Joint Conference on Neural Networks, IJCNN 2017, Anchorage, AK, USA, May 14-19, 2017*, 2017, pp. 888–896.
8. I. B. Yildiza, H. Jaeger, and S. J. Kiebel, "Re-visiting the echo state property," *Neural Networks*, vol. 35, pp. 1–9, 2012.
9. G. Manjunath and H. Jaeger, "Echo State Property Linked to an Input: Exploring a Fundamental Characteristic of Recurrent Neural Networks," *Neural Computation*, vol. 25, no. 3, pp. 671–696, March 2013.
10. M. Lukoševičius, "A Practical Guide to Applying Echo State Networks," in *Neural Networks: Tricks of the Trade*, ser. Lecture Notes in Computer Science, G. Montavon, G. Orr, and K.-R. Müller, Eds. Springer Berlin Heidelberg, 2012, vol. 7700, pp. 659–686. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-35289-8_36
11. N. Marwan, M. C. Romano, M. Thiel, and J. Kurths, "Recurrence plots for the analysis of complex systems," *Physics Reports*, vol. 438, pp. 237–329, 2007.
12. F. M. Bianchi, L. Livi, and C. Alippi, "Investigating Echo-State Networks Dynamics by Means of Recurrence Analysis," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 2, pp. 427–439, 2016.
13. Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *Neural Networks, IEEE Transactions on*, vol. 5, no. 2, pp. 157–166, 1994.
14. B. Schrauwen, M. Wardermann, D. Verstraeten, J. J. Steil, and D. Stroobandt, "Improving Reservoirs using Intrinsic Plasticity," *Neurocomputing*, vol. 71, pp. 1159–1171, March 2007.
15. J. J. Steil, "Backpropagation-Decorrelation: online recurrent learning with $O(N)$ complexity," in *Proceedings of IJCNN'04*, vol. 1, 2004.
16. Y. Xue, L. Yang, and S. Haykin, "Decoupled Echo State Networks with lateral inhibition," *Neural Networks*, vol. 20, no. 3, pp. 365–376, 2007.
17. H. Jaeger, M. Lukoševičius, D. Popovici, and U. Siewert, "Optimization and applications of Echo State Networks with leaky-integrator neurons," *Neural Networks*, vol. 20, no. 3, pp. 335–352, 2007.
18. J. Schmidhuber, D. Wierstra, M. Gagliolo, and F. Gomez, "Training Recurrent Networks by Evolino," *Neural Networks*, vol. 19, pp. 757–779, 2007.
19. S. Basterrech and G. Rubino, "Echo State Queueing Network: A new Reservoir Computing Learning Tool," in *10th IEEE Consumer Communications and Networking Conference, CCNC 2013, Las Vegas, NV, USA, January 11-14, 2013*, 2013, pp. 118–123. [Online]. Available: <http://dx.doi.org/10.1109/CCNC.2013.6488435>
20. A. Rodan and P. Tiño, "Minimum Complexity Echo State Network," *IEEE Transactions on Neural Networks*, vol. 22, pp. 131–144, 2011.