Echo State Networks for Multi-dimensional Data Clustering

Petia Koprinkova-Hristova[1] and Nikolay Tontchev[2]

[1] Bulgarian Academy of Sciences,
Institute of Information and Communication Technologies,
Acad. G. Bonchev str. bl.25A, 1113 Sofia, Bulgaria
pkoprinkova@bas.bg
[2] University of Transport,
158 Geo Milev str., bl.2, 1574 Sofia, Bulgaria
tontchev@vtu.bg

Abstract. In the present work we showed that together with improved stability the Intrinsic Plasticity (IP) tuned Echo State Network (ESN) reservoirs possess also better clustering abilities that opens a possibility for application of ESNs in multidimensional data clustering. The revealed ability of ESNs is demonstrated first on an artificially created data set with known in advance number and position of clusters. Automated procedure for multidimensional data clustering was proposed. It allows discovering multidimensional data structure without specification in advance the clusters number. The developed procedure was further applied to a real data set containing concentrations of three alloying elements in numerous steel compositions. The obtained number and position of clusters showed logical from the practical point of view data separation.

Keywords: Echo State Network, Intrinsic Plasticity, stability, data clustering.

1 Introduction

Designed to be as close as possible to biological brain structure artificial recurrent neural networks (RNNs) are a large and varied class [8]. Although they have a lot of applications, there are problems with their supervised training arising from their recurrent structure. Aimed at solving these problems, "reservoir computing" approaches aroused that greatly facilitated the practical application of RNNs [8]. The key idea is that a RNN (named reservoir) can be randomly generated and can remain unchanged during the training and running phases. The only trainable is the RNN readout that is linear combination of current reservoir neurons states and hence could be trained faster. An extensively investigated and developed branch of reservoir computing is called "Echo state network" (ESN) [3, 8]. It incorporates a randomly generated recurrent reservoir with sigmoid nonlinearities of neurons outputs (usually hyperbolic tangent). The only restriction is that such reservoir has to have "echo state property" meaning that the effect of its previous state and input to its output should vanish gradually in time, i.e. the reservoir possesses stable behavior. Since the ESN

A.E.P. Villa et al. (Eds.): ICANN 2012, Part I, LNCS 7552, pp. 571–578, 2012.
© Springer-Verlag Berlin Heidelberg 2012

reservoir structure is randomly generated, there are no universal recipes for its generation [8] and all works in this direction are task dependent. The usual recommendation for achieving the echo state property is to generate a reservoir weight matrix with spectral radius below one. However as it was mentioned in many works [8] this condition will not guaranty ESN stable behavior in general.

Since it is well known that any stable stationary state has a local maximum of entropy [4], it can be expected that maximization of entropy at the ESN reservoir output could be related to increasing of its stability. There are several works proposing ESN reservoir improvement related to entropy maximization [9, 11] and motivated by known biological mechanisms of changing neural excitability according to the distribution of the input stimuli [10]. In both cases it was proposed to use a bias term that will move the operating point of the system in the desired direction. In [10] the authors proposed a gradient method named Intrinsic Plasticity (IP) training for adjusting the biases as well as of an additional gain term aimed at achieving the desired distribution of reservoir output.

In our previous work [7] it was shown that in fact IP training achieves balance between maximization of entropy at the ESN reservoir output and its concentration around the pre-specified mean value. The simulation investigations with different random reservoirs showed that the IP improvement stabilizes even initially unstable reservoirs. Theoretical stability investigations showed that stabilization was achieved by squeezing of the neurons nonlinearities working sectors. During investigations why and how IP reservoir improvement influences its stability we observed another interesting effect: the reservoir neurons equilibrium points are not only moved but also are concentrated in several regions. Then question aroused: is it possible to use this effect for clustering purposes too? Indeed many well known RNNs used for data classification [1, 5, 6] relay on unsupervised learning procedures that minimize given energy function in search of correspondent to data structure adjustment of network equilibrium states. In [12] for the first time it was proposed to use ESN in image classification to “draw out” silent underlying features of the data. These extracted features were used further as inputs to a feedforward neural network classifier. Here we exploit the same reservoir ability but looking from another perspective: we consider combinations between steady states of each two neurons in the reservoir as numerous two-dimensional projections of the original multidimensional data fed into ESN input. These low dimensional projections can be used next for easier data clustering. Although the idea to apply designed for time series modeling ESN to static vectors clustering can appear odd, it is actually consistent with other dynamic neural architectures. As an example can be mentioned neural systems possessing multi-stable attractors that perform temporal integration aimed at discrimination between multiple alternatives [2].

The paper is constructed as follows: first the revealed effect of IP training is demonstrated on artificially created data sets whose number and position of clusters are known in advance; next an automated procedure for multidimensional data clustering was proposed that allows discovering structure of multidimensional data sets without specification in advance the clusters number; finally, in order to demonstrate capability of clustering of unclearly separated regions, the proposed procedure was applied to a real data set containing quantities of three alloying elements in different steel alloys types. The obtained number and position of clusters showed logical from the practical point of view data separation.

2 Problem Statement

2.1 Echo State Networks and IP Improvement of Reservoir

The ESN reservoir dynamics is described as follows [3]:

$$r(k)=f^{res}\left(W^{in}u(k)+W^{res}r(k-1)\right) \tag{1}$$

Here, $u(k)$ is the network input vector with size n_u, $r(k)$ - reservoir neurons states vector with size n_r; W^{in} and W^{res} are $n_r \times n_u$ and $n_r \times n_r$ matrices that are randomly generated and are not trainable. The neurons in the reservoir have a simple sigmoid output function f^{res} that is usually hyperbolic tangent.

The IP reservoir improvement proposed in [10, 11] is gradient descent procedure that minimizes the Kullback-Leibler divergence:

$$D_{KL}(p(r),p_d(r))=\int p(r)\log\left(\frac{p(r)}{p_d(r)}\right) \tag{2}$$

D_{KL} is a measure for the difference between the actual $p(r)$ and the desired $p_d(r)$ probability distribution of reservoir neurons output r. Since the commonly used transfer function of neurons is the hyperbolic tangent, the proper target distribution that maximizes the information at the output according to [10] is the Gaussian one. It is also recommended to use zero mean Gaussian distribution with desired variance σ so that IP training will tend to concentrate the reservoir outputs around chosen mean (i.e. zero) and squeeze it into the interval $[-3\sigma, 3\sigma]$. In order to achieve those effects two additional reservoir parameters - gain a and bias b (both vectors with n_r size) - are introduced as follows:

$$r(k)=f^{res}\left(diag(a)W^{in}u(k)+diag(a)W^{res}r(k-1)+b\right) \tag{3}$$

The IP training is procedure that adjusts vectors a and b using gradient descent.

2.2 Effect of IP Training on ESN Equilibrium States

The equilibrium state of reservoir r_e for a constant input u_c can be determined as follows:

$$r_e = \tanh\left(diag(a)W^{in}u_c + diag(a)W^{res}r_e + b \right) \tag{4}$$

So if $b=0$ and $u_c=0$, the equilibrium will be at the origin of the reservoir state space coordinate system. Otherwise it will be moved in dependence on the values of input u_c and bias b vectors. Since the input weights matrix remains constant, the first term in the brackets will be also constant for constant inputs (u_c =const.). Thus we can consider it together with the bias term as common bias for a given input vector:

$$r_e = \tanh\left(diag(a)W^{res}r_e + b_{u_c} \right), \quad b_{u_c} = diag(a)W^{in}u_c + b \tag{5}$$

Hence the reservoir equilibrium will be different for different input vectors. Moreover, if the input vectors are close in the input space, they will result in close equilibrium points in the reservoir state - a fact that could be exploited for clustering purposes. Since the connections matrices are randomly generated, the reservoir equilibrium states will vary randomly within the hyperbolic tangent output interval, i.e. [-1 +1]. After IP improvement they will be squeezed into the interval determined by chosen variance σ of Gaussian distribution, i.e. [-3 σ, +3 σ]. Hence we can suppose that overall IP training will probably lead to an ordered distribution of reservoir neurons equilibrium states.

3 Our Clustering Procedure

The above considerations motivated the experiment described below. It could be extended to multi-dimensional spaces but in order to be able to present results clearly our example is three dimensional one. Our experiment is as follows:

- Several clearly separated data clusters (shown on Figure 1 below) were generated in three dimensional unit cube space.
- Random ESN reservoir was generated and each data point was fed into its input many times while the reservoir achieves corresponding to this data equilibrium state.
- IP training procedure was applied by presenting all generated data and again the new reservoir equilibriums were determined for each data point from the data set.
- Two dimensional plots of all possible combinations between reservoir neurons equilibriums scaled within interval [-1 +1] were generated in order to see if there is any change of their positions.

Figure 2 presents an example of positions of several chosen neurons from our ESN reservoir before and after its IP tuning. It can be clearly seen that before IP training equilibrium points in two dimensional state spaces could not be clearly separated into different clusters. However, after IP training they appeared separable.

It is obvious that not all possible combinations of two neurons outputs give the same clear picture as can be seen from that figure. We can consider each couple of

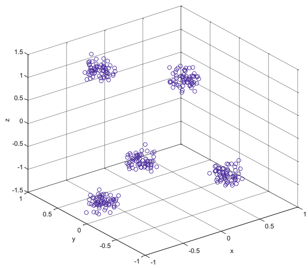

Fig. 1. Artificially generated five clusters data set

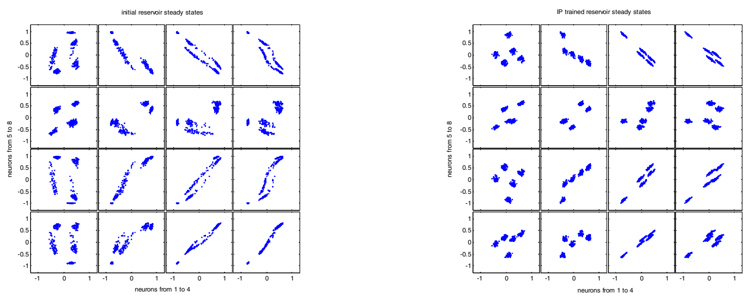

Fig. 2. Scaled equilibrium states of reservoir neurons before and after IP training

neurons as different "point of view projection" of multidimensional input space onto the two dimensional space of each couple of neurons equilibriums. However, question arises how to use this effect for clustering purposes?

Here we propose the following decision: since the IP training forces reservoir output to distribute according pre-specified Gaussian distribution, we decided to observe the obtained equilibrium states distributions. Left part of Figure 3 shows probability density distributions of equilibrium states of ten chosen neurons from our IP trained ESN reservoir. As it can be seen, each neuron output distribution is combination of several Gaussian distributions. Stars on the figure mark local maxima on the distribution curves that correspond to the local Gaussian distribution. Hence we can suppose that neurons with bigger number of maxima separate data into bigger number of clusters. So if we choose two dimensional projections formed by neurons with biggest number of probability distribution maxima, we can obtain clearest separation of data.

Here we propose the following algorithm for two dimensional projections choice:

- Calculate probability density distribution of all neurons equilibriums;
- Find number of local maxima of each probability density distribution;
- Choose neurons with higher number of local maxima in their probability density distributions;
- Choose two dimensional projections for all possible combinations between these selected neurons for data clustering;

Right side of the Figure 3 presents chosen by our algorithm best two dimensional projection from the numerous projections from Figure 2, i.e. projection obtained by

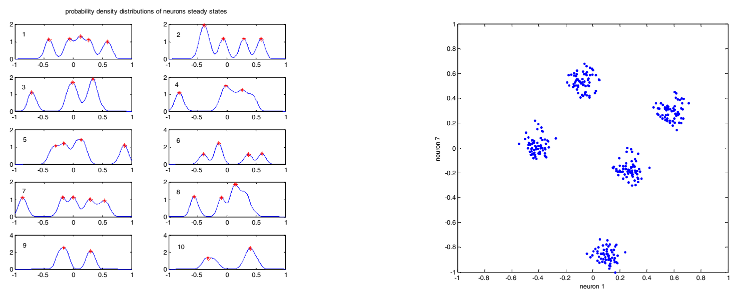

Fig. 3. Probability density distributions of several neurons in the reservoir (left) with local maxima marked by stars and chosen two dimensional projection (right)

neurons number 1 and 7 that have probability distributions with higher number of local maxima. As it is clearly seen, the chosen projection separates clearly five clusters of the original data set.

4 Real Data Clustering Experiment

Next we apply our clustering procedure to a real data set. It consists of 91 steel alloy compositions. Each data point contains concentrations of three main alloying elements: carbon (C), silicone (Si) and manganese (Mn) in percents (%). They can be separated naturally into three groups according to concentrations of Si and Mn. Tables 1 and 2 below summarize the data of the two smaller data groups. The rest of 91 data belong to the third biggest cluster.

Left part of Figure 4 presents all 91 data points in three dimensional space. The red squares correspond to the data from Table 1, the green circles – to the data from Table 2 and the blue circles – to the third data cluster.

Right part of the Figure 4 presents chosen by our procedure two dimensional projection of the data obtained after IP training of ESN and application of the proposed projection selection algorithm. The red (squares), blue (dots) and green (circles) marks correspond to the data from the three clusters separated in the left part of the figure.

Black stars (left) and squares (right) represent the clusters centers obtained by subtractive fuzzy clustering procedure. Original data are separated into 4 clusters while projected once – in 3 clusters that correspond better to the logical separation of our data set, although the red squares cluster center is moved towards the blue dots cluster due to restricted number of data in the red squares cluster.

Table 1. Class one (marked by 5 red squares): Mn≥1.6%

No	C, %	Si, %	Mn, %
1	0.35	0.27	1.6
2	0.45	0.27	1.6
3	0.305	0.27	1.6
4	0.36	0.27	1.75
5	0.4	0.27	1.6

Table 2. Class two (marked by 10 green circles): Si≥1.05

No	C, %	Si, %	Mn, %
1	0.355	1.25	0.95
2	0.41	1.4	0.45
3	0.2	1.05	0.95
4	0.315	1.05	0.95
5	0.38	1.2	0.45
6	0.25	1.05	0.95
7	0.31	1.05	0.95
8	0.33	1.2	0.45
9	0.305	1.05	1.15
10	0.34	1.25	0.9

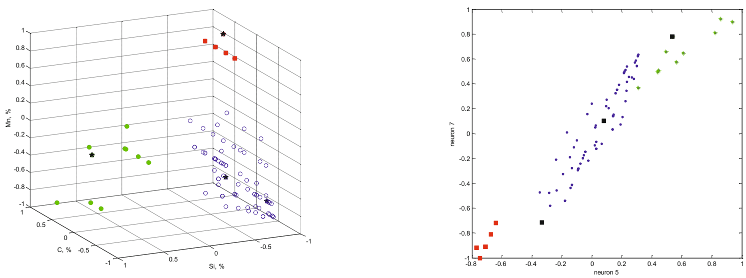

Fig. 4. Three-dimensional presentation of 91 steel compositions (left) and their separation into three clusters (right)

5 Conclusions

We showed experimentally that together with improved stability the Intrinsic Plasticity (IP) tuned Echo State Network (ESN) reservoirs possess also better clustering abilities that naturally opens the possibility to apply them for multidimensional data clustering. Based on investigated effect of IP improvement of ESN reservoir we propose a procedure for multidimensional data clustering. It allows discovering multidimensional data structure without specification in advance the clusters number. The developed procedure was applied also to a real data set containing different steels three alloying elements concentrations. The obtained number and position of clusters showed logical from the practical point of view data separation.

Our idea can easily be extended to more than 2-dimensional data projection by using more than two reservoir neurons. The idea of using neurons density distributions for projections discrimination needs further refinement since we observed strong dependence of obtained number of maxima on the bandwidth of used kernel-smoothing window. All these topics will be subject of our future work.

Acknowledgments. This work was partially supported by the Bulgarian National Science Fund under the Project No DDVU02/11.

References

1. Ackley, D.H., Hinton, G.E., Sejnowski, T.J.: A learning algorithm for Boltzmann machines. Cognitive Science 9, 147–169 (1985)
2. Brody, C.D., Romo, R., Kepecs, A.: basic mechanisms for graded persistent activity: Discrete attractors, continuous attractors, and dynamical representations. Current Opinion in Neurobiology 13, 204–211 (2003)
3. Jaeger, H.: Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the "echo state network" approach. GMD Report 159, German National Research Center for Information Technology (2002)
4. Haddad, W.M., Chellaboina, V.S., Nersesov, S.G.: Thermodynamics: A Dynamical System Approach. Princeton University Press (2005)
5. Hinton, G.E., Salakhutdinov, R.: Reducing the dimensionality of data with neural networks. Science 313(5786), 504–507 (2006)
6. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proc. of National Academy of Sciences USA 79, 2554–2558 (1982)
7. Koprinkova-Hristova, P., Palm, G.: ESN Intrinsic Plasticity versus Reservoir Stability. In: Honkela, T. (ed.) ICANN 2011, Part I. LNCS, vol. 6791, pp. 69–76. Springer, Heidelberg (2011)
8. Lukosevicius, M., Jaeger, H.: Reservoir computing approaches to recurrent neural network training. Computer Science Review 3, 127–149 (2009)
9. Ozturk, M., Xu, D., Principe, J.: Analysis and design of Echo state networks. Neural Computation 19, 111–138 (2007)
10. Schrauwen, B., Wandermann, M., Verstraeten, D., Steil, J.J., Stroobandt, D.: Improving reservoirs using intrinsic plasticity. Neurocomputing 71, 1159–1171 (2008)
11. Steil, J.J.: Online reservoir adaptation by intrinsic plasticity for back-propagation-decoleration and echo state learning. Neural Networks 20, 353–364 (2007)
12. Woodward, A., Ikegami, T.: A reservoir computing approach to image classification using coupled echo state and back-propagation neural networks. In: Proc. of 26th Int. Conf. on Image and Vision Computing, Auckland, New Zealand, November 29-December 1, pp. 543–548 (2011)