Physica A xx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

Q1 Minimal perceptrons for memorizing complex patterns

Q2 Marissa Pastor [a,b,1], Juyong Song [a,c,1], Danh-Tai Hoang [a,d], Junghyo Jo [a,c,*]

[a] Asia Pacific Center for Theoretical Physics, Pohang, South Korea
[b] Department of Physics, University of San Carlos Talamban, Cebu, Philippines
[c] Department of Physics, Pohang University of Science and Technology, Pohang, South Korea
[d] Department of Natural Sciences, Quang Binh University, Dong Hoi, Viet Nam

HIGHLIGHTS

- A new complexity measure for binary patterns is proposed.
- The measure estimates the minimal network size for memorizing binary patterns.
- The predicted minimal network size agrees with simulations of machine learning.

ARTICLE INFO

Article history:
Received 15 December 2015
Received in revised form 4 April 2016
Available online xxxx

Keywords:
Perceptrons
Network complexity
Binary patterns
Memory storage
Network architecture

ABSTRACT

Feedforward neural networks have been investigated to understand learning and memory, as well as applied to numerous practical problems in pattern classification. It is a rule of thumb that more complex tasks require larger networks. However, the design of optimal network architectures for specific tasks is still an unsolved fundamental problem. In this study, we consider three-layered neural networks for memorizing binary patterns. We developed a new complexity measure of binary patterns, and estimated the minimal network size for memorizing them as a function of their complexity. We formulated the minimal network size for regular, random, and complex patterns. In particular, the minimal size for complex patterns, which are neither ordered nor disordered, was predicted by measuring their Hamming distances from known ordered patterns. Our predictions agree with simulations based on the back-propagation algorithm.

1. Introduction

Neural signaling in synaptic networks motivated the early study of artificial neural networks, to recapitulate the learning capability of the brain. Their utility has expanded from that inception to serving as alternatives to conventional computers for input/output processing or as exemplars of parallel distributed processing, and they have been successfully applied to pattern classification [1,2] and memory storage [3–6]. Standard implementations of neural networks map inputs $\boldsymbol{x}^\mu$ to outputs z^μ through intermediate processing layers. The M input/output pairs, $\xi^\mu = (\boldsymbol{x}^\mu, z^\mu)$, form a *pattern*, $\boldsymbol{\xi} = \{\xi^1, \xi^2, \ldots, \xi^M\}$. Note that the pattern in this work refers a set of input/output pairs, not just inputs as usually defined in "pattern" classification problems. This input/output mapping can be achieved in two different ways: The neural network can either (i) learn the underlying rule for the mapping from some training pairs, or (ii) memorize the whole pattern of

* Corresponding author at: Asia Pacific Center for Theoretical Physics, Pohang, South Korea.
E-mail address: jojunghyo@apctp.org (J. Jo).

1 These authors contributed equally to this work.

http://dx.doi.org/10.1016/j.physa.2016.06.025
0378-4371/

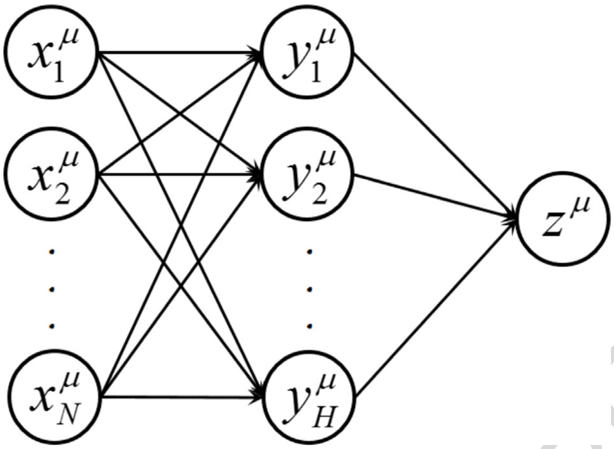

Fig. 1. Three-layer feedforward neural network.

input/output pairs and retrieve the stored outputs in response to given inputs. In either way, it is a major impediment that the required complexity of network architectures for learning or memorizing certain patterns is, in general, unknown. In this paper, we focus on memorizing patterns.

Designing the optimal network architecture has been known as an NP (Non-deterministic Polynomial-time) problem that requires computationally expensive search techniques and optimization [7–10]. Indeed, most attempts use empirical approaches and proceed by scanning over different network configurations while utilizing incremental [11] and/or pruning algorithms [12,13]. Simple networks may lead to insufficient memory and poor generalization, while complex networks lead to poor predictive performance by overestimating each element in patterns [6]. The required complexity of networks generally depends on the complexity of patterns for memorizing. Therefore, if the complexity of patterns and networks could be quantified, the optimal network architecture could be systematically designed.

Two popular complexity measures for patterns are Shannon entropy, the degree of uncertainty for describing a pattern [14–16], and Kolmogorov complexity, the length of the shortest computer program for generating a pattern [17]. However, the uncertainty or probability of each element in a pattern is unknown and the algorithmic complexity is itself difficult to compute. These difficulties suggest the need for a metric to quantify the practical complexity of patterns relevant for perceptrons. Here we propose a simple complexity index and relate it to the minimal network size for memorizing patterns.

This paper is organized as follows: We introduce the mathematical description of our feedforward neural network in Section 2, and a storage problem of binary patterns of different complexities in Section 3. Next, we estimate minimal network sizes for storing regular binary patterns in Section 4.1 and random binary patterns in Section 4.2, and compare them to simulation results. Then, we generalize the complexity formulation to estimate minimal network size for storing complex binary patterns in Section 5. Finally, we summarize the paper in Section 6.

2. Neural network

We study a three-layer feedforward neural network as shown in Fig. 1. This simple network architecture is successful in solving pattern recognition problems [18,19]. In addition, the universal approximation theorem proves that the three-layer network suffices to approximate any continuous function, $z^\mu = f(\boldsymbol{x}^\mu)$ [20]. For simplicity, we consider N-dimensional vectors of binary inputs $\boldsymbol{x}^\mu = (x_1^\mu, x_2^\mu, \ldots, x_N^\mu)$ and scalar binary outputs z^μ. One pattern $\boldsymbol{\xi}$ represents 2^N pairs of $(\boldsymbol{x}^\mu, z^\mu)$, because each component in the N-dimensional input vector takes values $x_i^\mu = 0$ or 1. The input/output mapping requires N input nodes and a single output node. In the feedforward three-layer network, an input $\boldsymbol{x}^\mu$ is transformed into the activities $\boldsymbol{y}^\mu = \{y_1^\mu, y_2^\mu, \ldots, y_H^\mu\}$ of H hidden nodes:

$$y_j^\mu = \sigma\Big(\sum_{i=1}^{N} w_{ji} x_i^\mu - w_{j0}\Big), \tag{1}$$

where w_{ji} is the connection weight from the ith input node to the jth hidden node, and w_{j0} is the bias of the jth hidden node. With these definitions, the jth hidden node is activated when the integrated input signal $\sum_i w_{ji} x_i^\mu$ exceeds the bias w_{j0} through the sigmoidal activation function, $\sigma(a) = 1/(1 + \mathrm{e}^{-a})$. The transformation from the hidden layer to the output layer follows the same rule:

$$z^\mu = \sigma\Big(\sum_{j=1}^{H} v_j y_j^\mu - v_0\Big), \tag{2}$$

Please cite this article in press as: M. Pastor, et al., Minimal perceptrons for memorizing complex patterns, Physica A (2016), http://dx.doi.org/10.1016/j.physa.2016.06.025