Neural Networks 94 (2017) 212–219

Contents lists available at ScienceDirect

Neural Networks

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

Recurrent networks with soft-thresholding nonlinearities for lightweight coding

MohammadMehdi Kafashan [a,c], ShiNung Ching [a,b,*]

[a] Department of Electrical and Systems Engineering, Washington University in St. Louis, One Brookings Drive, Campus Box 1042, MO 63130, United States
[b] Division of Biology and Biomedical Sciences, Washington University in St. Louis, One Brookings Drive, Campus Box 1042, MO 63130, United States
[c] Department of Neurobiology, Harvard Medical School, 220 Longwood Ave, Boston, MA 02115, United States

ARTICLE INFO

Article history:
Received 27 January 2017
Received in revised form 18 June 2017
Accepted 7 July 2017
Available online 22 July 2017

Keywords:
Efficient sparse coding
Unsupervised learning
Short-term memory
Proximal gradient descent
Neural networks

ABSTRACT

A long-standing and influential hypothesis in neural information processing is that early sensory networks adapt themselves to produce efficient codes of afferent inputs. Here, we show how a nonlinear recurrent network provides an optimal solution for the efficient coding of an afferent input and its history. We specifically consider the problem of producing lightweight codes, ones that minimize both ℓ_1 and ℓ_2 constraints on sparsity and energy, respectively. When embedded in a linear coding paradigm, this problem results in a non-smooth convex optimization problem. We employ a proximal gradient descent technique to develop the solution, showing that the optimal code is realized through a recurrent network endowed with a nonlinear soft thresholding operator. The training of the network connection weights is readily achieved through gradient-based local learning. If such learning is assumed to occur on a slower time-scale than the (faster) recurrent dynamics, then the network as a whole converges to an optimal set of codes and weights via what is, in effect, an alternative minimization procedure. Our results show how the addition of thresholding nonlinearities to a recurrent network may enable the production of lightweight, history-sensitive encoding schemes.

1. Introduction

1.1. Background

It has been hypothesized that the early stages of sensory neural processing have evolved to encode environmental signals with minimal consumption of biological resources (Attneave, 1954; Bialek, de Ruyter Van Steveninck, & Tishby, 2006; King, Zylberberg, & DeWeese, 2013; Laughlin, 2001; Olshausen et al., 1996), i.e., the notion of efficient neural coding (Baddeley et al., 1997; Doi et al., 2012; Graham, Chandler, & Field, 2006; Liu, Stevens, & Sharpee, 2009; Major & Tank, 2004; Schwartz & Simoncelli, 2001; Smith & Lewicki, 2006; Srinivasan, Laughlin, & Dubs, 1982). A popular model within efficient coding is that of sparse coding, which posits that sensory information is encoded using a small number of active neurons at any given point in time (Lee, Battle, Raina, & Ng, 2006; Mairal, Bach, Ponce, & Sapiro, 2009, 2010). This has, in particular, been considered a plausible model of visual cortical networks (Lee et al., 2006; Olshausen & Field, 1997, 2004). Specifically, it has been shown that sparse coding produces localized bases when applied to natural images (Olshausen, 2000) similar to those observed in biological neurons in visual cortex (Olshausen & Field, 2004). More generally, sparse coding can be applied to learning overcomplete basis sets (Lewicki & Sejnowski, 2000), in which the number of bases is greater than the input dimension, which contrasts unsupervised learning techniques such as principal component analysis. Additionally, neural networks have been used widely to approximate continuous functions (Barron, 1993; Costarelli, 2015; Costarelli & Spigler, 2015; Costarelli & Vinti, 2016a, c, d; Di Marco, Forti, Grazzini, & Pancioni, 2014; Gripenberg, 2003; Klusowski & Barron, 2016) in different applications such as manifold learning (Chui & Mhaskar, 2016) and image classification (Cao, Liu, & Park, 2013). In Costarelli and Vinti (2016b), Costarelli & Vinti established convergence properties of a specific form of such networks.

1.2. Sparse coding framework

The goal of minimal energy sparse coding is to efficiently represent time-varying input vectors approximately as a weighted linear combination of a small number of unknown basis vectors. These basis vectors capture high-level salient structure in the input data. Here, an m-dimensional input signal, denoted as $\mathbf{x}$ is encoded

* Corresponding author at: Department of Electrical and Systems Engineering, Washington University in St. Louis, One Brookings Drive, Campus Box 1042, MO 63130, United States.

E-mail addresses: kafashan@ese.wustl.edu (M. Kafashan), shinung@ese.wustl.edu (S. Ching).

http://dx.doi.org/10.1016/j.neunet.2017.07.008
0893-6080/© 2017 Elsevier Ltd. All rights reserved.

or represented using basis vectors $\mathbf{d}_1, \dots, \mathbf{d}_n \in \mathbb{R}^m$ and a sparse vector of weights or firing rates $\mathbf{r} \in \mathbb{R}^n$ as

$$\mathbf{x} \simeq \hat{\mathbf{x}} = \mathbf{D}\mathbf{r}. \tag{1}$$

The nominal goal is to find representations **r** that are 'lightweight', i.e., minimal with respect to some combination of ℓ_1 and ℓ_2 criteria. Mathematically, this can be formulated generally in terms of optimization of the following objective

$$J(\mathbf{r}) = \frac{1}{2} \|\mathbf{x} - \mathbf{D}\mathbf{r}\|_2^2 + \frac{\lambda_2}{2} \|\mathbf{r}\|_2^2 + \lambda_1 \|\mathbf{r}\|_1 \tag{2}$$

with respect to **r**. Here, the first term is the data fidelity term (the error between the reconstructed input signal and the actual stimulus), the second term corresponds to the 'energetic' cost of the signal representation, and the last term controls the sparsity level of the representation.

1.3. Prior results

Optimizing objectives of the form (2) can be formulated and solved in continuous and discrete frameworks. In the former, the idea is to find a (continuous) dynamical system, or network, that emits the optimal solution. In this regard, it has been shown that a continuous-time algorithm based on the principles of thresholding and local competition can solve a family of such problems (Rozell, Johnson, Baraniuk, & Olshausen, 2007, 2008). The derived system uses computational primitives that correspond to simple analog elements, making possible a parallel implementation. Further, global convergence to the true optimal solution has been established in Balavoine, Romberg, and Rozell (2012).

In the discrete framework, the dominant approach to sparse coding has utilized convex optimization techniques. Many algorithms have been employed in this context such as gradient projection (Nowak, Wright, et al., 2007), interior point method algorithms (Candes & Romberg, 2005; Kim, Koh, Lustig, Boyd, & Gorinevsky, 2007), and iterative thresholding methods (Bioucas-Dias & Figueiredo, 2007; Blumensath, Yaghoobi, & Davies, 2007).

Sussillo and Abbott developed a learning scheme called FORCE within the recurrent setting to construct networks that produce a wide variety of complex output patterns that require memory (Sussillo & Abbott, 2009). However, they use backpropagation to train the network which results in a non-local algorithm that is not biologically plausible. In contrast, there has been recent interest to show how simple forms of back-propagation could be realized in a biologically plausible setting (Schiess, Urbanczik, & Senn, 2016; Urbanczik & Senn, 2014). They present a simple compartmental neuron model together with a non-Hebbian, learning rule for dendritic synapses. It has been shown recently (DeWolf, Stewart, Slotine, & Eliasmith, 2016; Gilra & Gerstner, 2017) that adaptive control theory can be utilized to find local learning rules, though the developed networks may exhibit cost-inefficient spiking architectures. In Alemi, Machens, Denève, and Slotine (2017), the authors proposed a local learning rule in a spiking neural network with similar cost function as considered here (without the history term) for learning arbitrary complex dynamics. They build a network that learns to efficiently represent its inputs while expending the least amount of spikes.

1.4. Contributions

In this paper, we consider the problem of discrete-time sparse coding with cost of the form (2), but with an additional objective related to encoding the input history, i.e., to also enable reconstruction via:

$$\mathbf{x}(t-q) \simeq \mathbf{D}\mathbf{S}^q\mathbf{r}(t), \tag{3}$$

where q is a positive integer and **S** is a history-decoding matrix. We specifically seek to solve this problem by means of constructing a (discrete-time) dynamical network of locally acting nodes that takes an input **x** and subsequently produces **r** as an output. In other words, we are interested in networks which can store memory of the ongoing network activity. Secondarily, we consider the problem of simultaneously learning both **r** and also the decoding matrices **D** and **S**.

The problem of developing memory or history-sensitive codes is itself well-studied. For example, it has been examined without sparsity constraints using backpropagation through time in Hochreiter and Schmidhuber (1997) and Werbos (1990). However, such a solution technique is not amenable to implementation in terms of a dynamical network. Other works have studied the short-term memory in linear echo state networks with random recurrent connections (Jaeger, 2002; LukošEvičIus & Jaeger, 2009; Maass, Natschläger, & Markram, 2002). In White, Lee, and Sompolinsky (2004), the authors reported that in the presence of noise, a particular class of orthogonal networks could have memory capacity that scales with network size. Further, it has been shown in Vertechi, Brendel, and Machens (2014) that a linear recurrent neural network can learn to efficiently represent both its present and past inputs with local learning rules for network connections. More recently, the problem of memory encoding has been treated with overt sparsity constraints (Charles, Yap, & Rozell, 2014; Ganguli & Sompolinsky, 2010) using ℓ_1 minimization methods over a receding horizon of a scalar-valued input signal. However, these results pertain primarily to sparsity of the input in time, i.e., many samples are zero, as opposed to sparsity within the vectors **x** or **r**.

The specific contributions of this paper are as follows:

1. We consider the discrete-time optimization of (2) with the additional objective of encoding input history. To handle the non-smoothness of the objective, we employ the proximal gradient descent method (Bauschke, Goebel, Lucet, & Wang, 2008; Chen et al., 2012; Parikh & Boyd, 2013) and subsequently derive a two-layer, nonlinear soft-thresholding network that generates the optimal solution. The network resembles the continuous time analogs referenced above, but with the generalization of encoding history.
2. We propose an online local adaptation rule that enables the simultaneous learning of the network weights (i.e., related to **D** and **S**). This adaptation occurs on a slower time-scale that the convergence of **r**, resulting in a network that, in essence, performs an alternative minimization procedure.
3. We provide several numerical examples that illustrate the performance of the proposed network in encoding input and history, highlighting in particular dependence on network size.

In our approach, we combine predictive coding with local learning rules for the connections in the network. Recently, Thalmeier and colleagues (Thalmeier, Uhlmann, Kappen, & Memmesheimer, 2016) used a similar strategy, combining predictive coding with FORCE learning (Sussillo & Abbott, 2009), and showed that their developed network could learn tasks such as the generation of desired chaotic activity. In this approach, the network is trained by propagating the error between the network output and a desired reference signal. The reliance on feedback of an error signal in this fashion results in non-local learning rules. In contrast, our framework is unsupervised and uses only local learning rules. Our proposed network is also different from echo-state networks where the recurrent connections are fixed, and only the connections in the output layer are updated (LukošEvičIus & Jaeger, 2009; Maass et al., 2002). In our case, all connections, including recurrent ones, adapt over time.