Neural Networks 105 (2018) 419–430

Contents lists available at ScienceDirect

Neural Networks

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

Exploiting layerwise convexity of rectifier networks with sign constrained weights

Senjian An [a],*, Farid Boussaid [b], Mohammed Bennamoun [a], Ferdous Sohel [c]

[a] *Department of Computer Science and Software Engineering, The University of Western Australia, Australia*
[b] *Department of Electrical, Electronic and Computer Engineering, The University of Western Australia, Australia*
[c] *School of Engineering and Information Technology, Murdoch University, Australia*

ARTICLE INFO

Article history:
Received 14 November 2017
Received in revised form 29 May 2018
Accepted 5 June 2018
Available online 13 June 2018

Keywords:
Rectifier neural network
Geometrically interpretable neural network
The majorization–minimization algorithm

ABSTRACT

By introducing sign constraints on the weights, this paper proposes sign constrained rectifier networks (SCRNs), whose training can be solved efficiently by the well known majorization–minimization (MM) algorithms. We prove that the proposed two-hidden-layer SCRNs, which exhibit negative weights in the second hidden layer and negative weights in the output layer, are capable of separating any number of disjoint pattern sets. Furthermore, the proposed two-hidden-layer SCRNs can decompose the patterns of each class into several clusters so that each cluster is convexly separable from all the patterns from the other classes. This provides a means to learn the pattern structures and analyse the discriminant factors between different classes of patterns. Experimental results are provided to show the benefits of sign constraints in improving classification performance and the efficiency of the proposed MM algorithm.

1. Introduction

In recent years, deep neural networks have achieved outstanding performance in various applications such as object recognition (He, Zhang, Ren, & Sun, 2015; Huang, Liu, Weinberger, & van der Maaten, 2017; Krizhevsky, Sutskever, & Hinton, 2012; Lee, Xie, Gallagher, Zhang, & Tu, 2015; Zeiler & Fergus, 2014), face verification (Sun, Chen, Wang, & Tang, 2014; Taigman, Yang, Ranzato, & Wolf, 2014), speech recognition (Deng et al., 2013; Hinton et al., 2012; Seide, Li, & Yu, 2011) and handwritten digit recognition (Ciresan, Meier, & Schmidhuber, 2012). These practical successes of deep neural networks have fuelled increased research into the optimization theory of neural networks, and many theoretical works have been reported to address questions, such as why local search methods as simple as gradient-based methods can train deep neural networks successfully, despite the inherent nonconvexity of the associated optimization problem. Both encouraging and discouraging results have been reported. For shallow neural networks with one hidden layer, it has been shown that, if the network is over-parameterized, and large enough compared to the data size, then there are no bad local minima (Boob & Lan, 2017; Haeffele & Vidal, 2015; Livni, Shalev-Shwartz, & Shamir, 2014; Nguyen & Hein, 2017; Poston, Lee, Choie, & Kwon, 1991; Soltanolkotabi, Javanmard, & Lee, 2017; Soudry & Carmon, 2016). For deep neural networks, Kawaguchi (2016) shows that there is no bad local minima during the training of deep linear networks, wherein the activation function is linear. For nonlinear activation functions, such as rectifier and max pooling, when the activation patterns of all the training data are fixed, deep nonlinear networks reduce to deep linear networks, and thus local minima do not exist either. However, these encouraging results are either under unreasonable assumptions or limited to shallow neural networks with one hidden layer. On the other hand, it has been shown that local minima occur commonly even for the simplest single-hidden-layer rectifier neural networks (Safran & Shamir, 2017) when minimizing the expected loss of inputs with a Gaussian distribution, or even with a single neuron (Auer, Herbster, & Warmuth, 1996) when minimizing the average loss over some arbitrary finite dataset. These discouraging results imply that, in general, local minima do exist for the optimization of neural networks. To explain the success of gradient methods in training deep neural networks, further research is required to find reasonable conditions under which bad local minima do not exist or the risk of being stuck in bad local minima is not severe. Recent research has discovered that some non-convex optimization problems, in machine learning, do not have bad local minima under reasonable assumptions (Bhojanapalli, Neyshabur, & Srebro, 2016; Ge, Huang, Jin, & Yuan, 2015; Ge, Lee, & Ma, 2016; Sun, Qu, & Wright, 2015), but for neural networks, the reasonable conditions are yet to be found. To find such conditions, the investigation of local convexity properties such as the layerwise convexity and the piecewise convexity might be required. A recent work (Ristera & Rubin, 2017) has investigated the training of rectifier neural networks using the

* Corresponding author.
E-mail addresses: senjian.an@uwa.edu.au, senjian.an@gmail.com (S. An).

https://doi.org/10.1016/j.neunet.2018.06.005
0893-6080/© 2018 Elsevier Ltd. All rights reserved.

piecewise convexity property of the objective functions. It proved that, when the objective functions are convex in the output layer of rectifier neural networks, they are piecewise convex functions of the parameters of each layer when the other parameters are fixed. However, there is an exponentially large number of pieces, for which the objective function is convex within each piece but may not be convex across all the pieces.

Since local minima may be encountered when training conventional neural networks (Auer et al., 1996; Safran & Shamir, 2017), this paper presents a new type of rectifier neural network, whose cost function has layer-wise convex bounds so that the local minima risk can be reduced using the well-known majorization–minimization (MM) algorithm (Sun, Babu, & Palomar, 2017). In the proposed two-hidden-layer sign constrained rectifier networks (SCRNs), the weights of the second hidden layer and those of the output layer are constrained to be non-positive. Despite these constraints, this type of neural network is still capable of separating any number of disjoint pattern sets (Section 4). When the sum of hinge loss and a convex regularization term are used as the cost function to train the proposed neural networks, the cost function can be minimized using the MM algorithm, which is an iterative optimization method exploiting partial convexities of a function in order to avoid bad local minima. The MM algorithm operates by finding a convex surrogate function which upperbounds the objective function. Optimizing the surrogate function drives the objective function downwards until a local optimum is reached. For the training of SCRNs, we show that, with any initialization of the parameters, there is a surrogate function that is convex as a function of each layer's parameters when all the other parameters are fixed. Hence, each layer's weights and biases can be learnt alternatively using the MM algorithm. Furthermore, SCRNs can also decompose each pattern set into several clusters so that each cluster is convexly separable from the patterns of the other classes (Section 4). They can thus be used to learn the pattern structures and analyse the discriminant factors between the patterns of different classes. These techniques enable feature analysis for knowledge discovery and for manual supervision to improve the efficiency and performance in training the classifiers. Typical applications include: (i) Feature discovery—In health and production management of precision livestock farming (Wathes, Kristensen, Aerts, & Berckmans, 2008), one needs to identify the key features associated with diseases (e.g. hock burn of broiler chickens) on commercial farms, using routinely collected farm management data (Hepworth, Nefedov, Muchnik, & Morgan, 2012); (ii) Supervised shape-free clustering for knowledge discovery—The proposed SCRNs can be used to separate each class of patterns into several clusters (i.e., convex subsets) so that each cluster of the patterns is convexly separable from other classes of patterns, wherein the clusters are not required to be of any particular shape other than convex polytopes; (iii) Human-supervised neural network training—The proposed two hidden-layer SCRNs transform the input data into convexly separable data using the first hidden layer. They further transform the data into linearly separable data using the second hidden layer. The decomposition properties of the SCRNs enable human to visualize the patterns, identify the outliers, check the separating boundaries and supervise the training by removing the outliers or mislabelled data.

Main contributions. In summary, the main contributions of this paper include:

- The introduction of sign constraints on the weights of neural networks in order to learn geometrically-interpretable models (Sections 2–4). When sign constraints are imposed on the weights of the proposed SCRNs, the first hidden layer transforms the data to be convexly separable, while the second hidden layer further transforms the data to be linearly separable. Consequently, every node is a concave (or convex) function of the weights of the preceding hidden layer. Since a concave (or convex) piecewise linear function is the minimum (or the maximum respectively) of several linear functions, the separating boundaries of the learnt SCRN classifiers are thus the union of several hyperplanes. This improves the geometrical interpretability of the classifiers and can be used to analyse the discriminant features between different classes of patterns. Our experimental results (Section 6) demonstrate that the learnt convex model, through a sign constrained neural network, can be well approximated by the minimum of several linear classifiers in the feature space of the second last hidden layer. This property can be used to analyse the key features of the learnt classifiers.
- Sign constraints induce sparsity and improve classification accuracy. Sign constraints move negative weights to be zero and thus some weights of the learnt neural networks are zero. This results in learning sparse neural networks, which has potential to improve classification accuracy. The experimental results provided in Section 6 (Table 1) demonstrate that sign constraints consistently improve performances across different neural networks and across validation and testing sets of the data.
- The introduction of MM algorithms for the training of sign constrained rectifier neural networks (Section 5). The convexity/concavity properties of the proposed SCRNs result in the existence of a convex surrogate function to upperbound the non-convex hinge loss function so that the efficient MM algorithm can be used to learn the parameters of the neural networks. Experimental results (Section 6) demonstrate that the proposed MM algorithm converges within a few iterations, while the gradient descent training of a conventional neural network usually takes thousands of iterations.

Table 1
Performance (error rate) comparison between neural networks with sign constraints (SC) and those without constraints. BL stands for the baseline neural network, while BN stands for the neural network with batch normalization layers.

	Training	Validation	Testing
BL	0%	0.88%	0.91%
BL+SC	0%	**0.81%**	**0.81%**
BN	0%	0.80%	0.85%
BN+SC	0%	**0.77**%	**0.75**%

Related works. This work is related to Ristera and Rubin, (2017) which exploits piecewise convexity properties of rectifier neural networks to overcome local minima problems. While Ristera and Rubin (2017) use the piecewise convexity of general rectifier neural networks, this work introduces layer-wise convexity/concavity properties by imposing sign constraints on the weights of the networks, and exploits these properties for pattern decomposition and for efficient training using MM algorithms to reduce the risk of bad local minima. This work on the universal classification power is related to Hornik, Stinchcombe, and White (1989), Le Roux and Bengio (2010) and Montufar and Ay (2011), which address the universal approximation power of deep neural networks for functions or for probability distributions, and An, Boussaid, and Bennamoun (2015) which prove that any multiple pattern sets can be transformed to be linearly separable by two hidden layers, with additional distance preserving properties. In this paper, we prove that any number of pattern sets can be separated by a three-layer (two hidden and one output) neural network with negative weights in the output layer and negative weights in the second hidden layer. The biases and the weights in the first hidden layer can either be positive or negative. The significance of the proposed