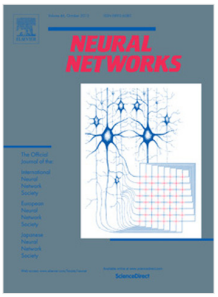
Accepted Manuscript



Exploiting layerwise convexity of rectifier networks with sign constrained weights

Senjian An, Farid Boussaid, Mohammed Bennamoun, Ferdous Sohel

PII: S0893-6080(18)30187-4
DOI: https://doi.org/10.1016/j.neunet.2018.06.005
Reference: NN 3971

To appear in: *Neural Networks*

Received date : 14 November 2017
Revised date : 29 May 2018
Accepted date : 5 June 2018

Please cite this article as: An, S., Boussaid, F., Bennamoun, M., Sohel, F., Exploiting layerwise convexity of rectifier networks with sign constrained weights. *Neural Networks* (2018), https://doi.org/10.1016/j.neunet.2018.06.005

Exploiting Layerwise Convexity of Rectifier Networks with Sign Constrained Weights

Senjian An[1], Farid Boussaid[2], Mohammed Bennamoun[1] and Ferdous Sohel[3]

[1]*School of Computer Science and Software Engineering*
The University of Western Australia
[2]*School of Electrical, Electronic and Computer Engineering*
The University of Western Australia
[3]*School of Engineering and Information Technology*
Murdoch University, Australia

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.

Keywords:
Rectifier Neural Network, Geometrically Interpretable Neural Network, The Majorization-Minimization Algorithm.

1. Introduction

In recent years, deep neural networks have achieved outstanding performance in various applications such as object recognition [1, 2, 3, 4, 5], face verification [6, 7], speech recognition ([8, 9, 10] and handwritten digit recognition [11]. 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 non-convexity 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 [12, 13, 14, 15, 16, 17, 18]. For deep neural networks, [19] 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 [20] when minimizing the expected loss of inputs with a Gaussian distribution, or even with a single neuron [21] 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 of 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 [22, 23, 24, 25]), 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 [26] has investigated the training of rectifier neural networks using the 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 [20,21], this paper presents a new type of rectifier neural network, whose cost function has layerwise convex bounds so that the local minima risk can be reduced using the well-known majorization-minimization (MM)

algorithm[27]. 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 (Sec. 4). When the sum of hinge loss and a convex regularisation term is used as the cost function to train the proposed neural networks, the cost function can be minimized using 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 downward 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 (Sec. 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 [28], 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 [29]; 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** (Sec. 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 interpretablity of the classifiers and can be used to analyse the discriminant features between different classes of patterns. Our experimental results (Sec. 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 Sec. 6 (Table 1) demonstrate that sign constraints consistently improve performances across different 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** (Sec. 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 (Sec. 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.

Related Works: This work is related to [26] which exploits piecewise convexity properties of rectifier neural networks to overcome local minima problems. While [26] uses 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 [30, 31, 32], which address the universal approximation power of deep neural networks for functions or for probability distributions, and [33] which proves 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 SCRNs lies in the fact that it can decompose each class of the patterns into several subsets, where each subset is convexly separable from the other classes of patterns. This decomposition can be used to

analyse pattern sets and identify the discriminant features for pattern recognition. Preliminary results of this paper were reported in [34]. This paper extends [34] to multi-category classification and presents MM-based efficient training algorithms for the proposed SCRNs.

Although the focus of this paper is static neural networks [35] where the input-output relationship is a static functional mapping, sign-constrained neural networks may also be useful for dynamic neural networks [36, 37, 38, 39, 40, 41, 42, 43, 44] for feedback control systems, where many of the feedback controllers are static functions of the states. As long as the controller is a static function of the state, the proposed rectifier network can be used to approximate the controller. Examples of such static controllers include the controller in Eq.(4) of [42] for sample data neural network based control systems, the controllers in Eq.(28) of [43] and Eq.(16) of [44] for sliding mode control systems. The convexity properties of the proposed sign-constrained rectifier networks might be useful in analysing the functions of nonlinear feedback control systems.

Notation. Throughout this paper, we use capital letters to denote matrices, lower case letters for scalar terms, and bold lower letters for vectors. For any integer m, we use $[m]$ to denote the integer set from 1 to m, i.e., $[m] \triangleq \{1, 2, \cdots, m\}$. $W \succeq 0$ and $\mathbf{b} \succeq 0$ denote that all elements of W and $\mathbf{b}$ are non-negative while $W \preceq 0$ and $\mathbf{b} \preceq 0$ denote that all elements of W and $\mathbf{b}$ are non-positive. Given a finite number of points $\mathbf{x}_i$ ($i \in [m]$) in $\mathbb{R}^n$, a convex combination $\mathbf{x}$ of these points is a linear combination of these points, in which all coefficients are non-negative and sum to 1. The convex hull of a set $\mathcal{X}$, denoted by $\mathrm{CH}(\mathcal{X})$, is a set of all convex combinations of the points in $\mathcal{X}$.

Organization. The rest of this paper is organised as follows. We introduce rectifier neural networks with sign constrained weights in Section 2, and investigate the capacity of sign constrained single hidden layer rectifier neural networks for classification and pattern decomposition in Section 3. Section 4 investigates the universal classification power and pattern decomposition capacity of two hidden layer rectifier neural networks with sign constraints on the output layer and on the last hidden layer. Such sign constraints can be used to control the strategy how a two-hidden-layer neural network to achieve linear separability. In Section 5, we first introduce the general MM algorithm, and then presents how MM algorithms can be used to train sign constrained neural networks. Experimental results are reported in Section 6 to demonstrate the benefits of sign constraints and the efficiency of the proposed MM algorithm. Section 7 concludes this paper.

2. Rectifier Neural Networks with Sign Constrained Weights

A hidden layer in a rectifier neural network can be described by a rectified linear unit (ReLU) as below

$$\mathrm{ReLU}(\mathbf{x}; W, \mathbf{b}) \triangleq \max(\mathbf{0}, W^T\mathbf{x} + \mathbf{b}) \tag{1}$$

where W is the weight matrix and $\mathbf{b}$ is the bias vector. $\mathrm{ReLU}(\mathbf{x}; W, \mathbf{b})$ is a simple yet powerful nonlinear transformation wherein the nonlinearity is imposed by the simplest nonlinear function $\max(0, x)$.

A rectifier neural network with m hidden layers can be described as a chain of m ReLUs.

$$\text{(ReNN)} \quad \begin{cases} \mathbf{z}_1 & \triangleq & \mathrm{ReLU}(\mathbf{x}; W_1, \mathbf{b}_1) \\ \mathbf{z}_k & \triangleq & \mathrm{ReLU}(\mathbf{z}_{k-1}; W_k, \mathbf{b}_k), 2 \le k \le m \\ \mathbf{y} & \triangleq & A^T\mathbf{z}_m + \mathbf{c} \end{cases} \tag{2}$$

where $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}_k$ are the input, the final output and the output of the k^{th} hidden layer respectively.

In particular, this paper considers a special class of rectifier neural networks with sign constraints on the weights of the output layer and the last hidden layer. The sign constraints are used to decompose the pattern sets for discriminate factor analysis.

A single hidden layer sign constrained ReNN imposes non-positiveness on the weights in the output layer and is defined as below:

$$\text{(SCReNN1)} \quad \begin{cases} \mathbf{z} & \triangleq & \mathrm{ReLU}(\mathbf{x}; W, \mathbf{b}) \\ \mathbf{y} & \triangleq & A^T\mathbf{z} + \mathbf{c} \\ A & \preceq & 0 \end{cases} \tag{3}$$

where $\mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{R}^m$ are the input and the output respectively, $W \in \mathbb{R}^{n\times l}, A \in \mathbb{R}^{l\times m}$ are the weight matrices and $\mathbf{b} \in \mathbb{R}^l, \mathbf{c} \in \mathbb{R}^m$ are the bias vectors in the hidden layer and output layer respectively.

For two hidden layer sign constrained ReNNs, we impose non-negativeness on the weights of the output layer and impose non-positiveness on the weights of the second hidden layer. A two hidden layer sign constrained ReNN can be described as below.

$$\text{(SCReNN2)} \quad \begin{cases} \mathbf{z}_1 & \triangleq & \mathrm{ReLU}(\mathbf{x}; W_1, \mathbf{b}_1) \\ \mathbf{z}_2 & \triangleq & \mathrm{ReLU}(\mathbf{z}_1; W_2, \mathbf{b}_2) \\ \mathbf{y} & \triangleq & A^T\mathbf{z}_2 + \mathbf{c} \\ W_2 & \preceq & 0 \\ A & \preceq & 0 \end{cases} \tag{4}$$

where $\mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{R}^n$ are the input and output , $W_1 \in \mathbb{R}^{n\times l_1}, W_2 \in \mathbb{R}^{l_1\times l_2}, A \in \mathbb{R}^{l_2\times m}$ are the weight matrices and $\mathbf{b}_k \in \mathbb{R}^{l_k} (k = 1, 2), \mathbf{c} \in \mathbb{R}^m$ are the bias vectors in the first hidden layer, the second hidden layer and the output layer.

Next, we present the properties of sign constrained rectifier networks. A real valued function $f(\mathbf{x})$ from $\mathbb{R}^n$ to $\mathbb{R}$ is called a *convex* function if

$$f(\lambda\mathbf{x}_1 + (1-\lambda)\mathbf{x}_0) \le \lambda f(\mathbf{x}_1) + (1-\lambda) f(\mathbf{x}_0) \tag{5}$$

holds for any $\lambda \in [0, 1], \mathbf{x}_1, \mathbf{x}_0 \in \mathbb{R}^n$. A real-valued function $f(\mathbf{x})$ is called concave if its negative $-f(\mathbf{x})$ is convex. The following lemma addresses the relationship between the convexity (or concavity) of a single hidden layer rectifier network and the signs of its weights in the output layer.

Lemma 1. *Let $f(\mathbf{x}; \mathbf{a}, W, \mathbf{b}, c) = \mathbf{a}^T \max\{0, W^T\mathbf{x} + \mathbf{b}\} + c$ be a real-valued function from $\mathbb{R}^n$ to $\mathbb{R}$. Then the following statements are true:*

(i) When the parameters $\mathbf{a}, W, \mathbf{b}, c$ *are fixed,* $f(\mathbf{x}; \mathbf{a}, W, \mathbf{b}, c)$ *is convex as a function of* $\mathbf{x}$ *if* $\mathbf{a} \succeq 0$*, and is concave if* $\mathbf{a} \preceq 0$*.*

(ii) When $\mathbf{x}, \mathbf{a}$ *are fixed,* $f(\mathbf{x}; \mathbf{a}, W, \mathbf{b}, c)$ *is convex as a function of* W *and* $\mathbf{b}$ *if* $\mathbf{a} \succeq 0$*, and is concave if* $\mathbf{a} \preceq 0$*.*

Proof: We only need to prove the first statement (i). The proof of (ii) is similar and thus omitted. Denote $\mathbf{a} = [a_1, a_2, \cdots, a_m]$, $W = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_m]$ and $\mathbf{b} = [b_1, b_2, \cdots, b_m]$, then

$$f(\mathbf{x}) = \sum_{i=1}^{m} a_i \max\{0, \mathbf{w}_i^T \mathbf{x} + b_i\} + c. \tag{6}$$

Let $\mathbf{x}_0$ and $\mathbf{x}_1$ be two points in Ω, i.e., $f(\mathbf{x}_0) > 0$ and $f(\mathbf{x}_1) > 0$, and let $\mathbf{x}_\lambda = \lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_0$. Denote $z_i(\lambda) = \mathbf{w}_i^T \mathbf{x}_\lambda + b_i$. Note that

$$\max\{0, z_i(\lambda)\} \leq \lambda \max\{0, z_i(0)\} + (1-\lambda)\max\{0, z_i(1)\} \tag{7}$$

holds for any $\lambda \in [0, 1]$. When $a_i \geq 0$ for any $i \in [m]$, we have

$$\begin{aligned} f(\mathbf{x}_\lambda) &= \sum_{i=1}^{m} a_i \max\{0, z_i(\lambda)\} + c \\ &\leq \lambda \sum_{i=1}^{m} a_i \max\{0, z_i(1)\} \\ &\quad + (1-\lambda) \sum_{i=1}^{m} a_i \max\{0, z_i(0)\} + c \\ &= \lambda f(\mathbf{x}_1) + (1-\lambda) f(\mathbf{x}_0) \end{aligned} \tag{8}$$

for any $\lambda \in [0, 1]$, and this implies that $f(\mathbf{x})$ is convex when $\mathbf{a} \succeq 0$.

□

From Lemma 1, we have the following two corollaries for the properties of sign constrained rectifier networks.

Corollary 2. *Let* SCReNN1 *be defined as in Eq. (3). Then every element of the output* $\mathbf{y}$ *is a concave function of* $\mathbf{x}$*.*

Corollary 3. *Let* SCReNN2 *be defined as in Eq. (4). Then every element of the output* $\mathbf{y}$ *is a concave function of* $\mathbf{z}_1$ *(i.e. the output of the first hidden layer).*

There are two useful properties of convex/concave functions. First, when a classifier is a convex/concave function, it separates the domain into two regions with one being a convex set: the set $\{\mathbf{x} : f(\mathbf{x}) < 0\}$ is a convex set when $f(\mathbf{x})$ is a convex function, and the set $\{\mathbf{x} : f(\mathbf{x}) > 0\}$ is a convex set if $f(\mathbf{x})$ is concave. Second, a convex/concave function can be approximated by the minimum/maximum of a number of linear classifiers. These two properties make the sign-constrained rectifier networks more geometrically interpretable. In Section 3 and Section 4, we will investigate how to use these properties to decompose the data for discriminant factor analysis.

3. The Capacity of Sign Constrained Rectifier Neural Networks with Single Hidden Layers

The complexity of pattern recognition problems can be quite different in practice. In Section 3.1, We first examine the different categories of classification problems based on the complexities of the patterns' separating boundaries. Then, we investigate the capacity of single hidden layer nets in Section 3.2.

3.1. Separability of Pattern Sets

Let $\mathcal{X}_1, \mathcal{X}_2$ be two disjoint pattern sets, that is, $\mathcal{X}_1 \cap \mathcal{X}_2 = \emptyset$. We introduce the following three categories of binary classification problems.

Linear-Separability of Two Categories: We say $\mathcal{X}_1$ is linearly-separable from $\mathcal{X}_2$ if there is a hyperplane in the vector space to separate $\mathcal{X}_1$ from $\mathcal{X}_2$. Note that, if $\mathcal{X}_1$ is linearly-separable from $\mathcal{X}_2$, then $\mathcal{X}_2$ is also linearly-separable from $\mathcal{X}_1$, so linear-separability is mutual. It is also known that, $\mathcal{X}_1$ and $\mathcal{X}_2$ are linearly-separable if and only if $\mathrm{CH}\{\mathcal{X}_1\} \cap \mathrm{CH}\{\mathcal{X}_2\} = \emptyset$.

Unidirectional Convex-Separability of Two Categories: $\mathcal{X}_1$ is called convexly-separable from $\mathcal{X}_2$ if there is a convex region including all the points in $\mathcal{X}_1$ while excluding all the points in $\mathcal{X}_2$. $\mathcal{X}_1$ is convexly-separable from $\mathcal{X}_2$ if and only if $\mathrm{CH}\{\mathcal{X}_1\} \cap \mathcal{X}_2 = \emptyset$.

Mutual Convex-Separability of Two Categories: $\mathcal{X}_1$ and $\mathcal{X}_2$ are called mutually convexly-separable from each other if each of them is convexly-separable from the other. They are mutually convexly-separable if and only if $\mathrm{CH}\{\mathcal{X}_1\} \cap \mathcal{X}_2 = \emptyset$ and $\mathrm{CH}\{\mathcal{X}_2\} \cap \mathcal{X}_1 = \emptyset$. Note that mutual convex-separability is weaker than linear separability, that is, any two linearly-separable pattern sets are mutually convexly-separable but mutually convexly-separable pattern sets may not be linearly-separable.

3.1.1. Separability of Multiple Pattern Sets

For multiple category data sets $\mathcal{X}_1, \mathcal{X}_2, \cdots, \mathcal{X}_m$, an m-dimensional function is usually used to classify the patterns. We call $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_m(\mathbf{x})]$ an $m-$dimensional separator of m disjoint pattern sets $\{\mathcal{X}_k, k \in [m]\}$ if, for each $k \in [m]$, $f_k(\mathbf{x}) > 0$ for all $\mathbf{x} \in \mathcal{X}_k$ and $f_k(\mathbf{x}) < 0$ for all $\mathbf{x} \in \bigcup_{j \neq k} \mathcal{X}_j$.

Pairwise Linear-Separability of Multiple Categories: We say m pattern sets $\{\mathcal{X}_i\}_{i=1}^m$ are pairwise linearly-separable if every two pattern sets are linearly-separable, that is,

$$\mathrm{CH}\{\mathcal{X}_i\} \cap \mathrm{CH}\{\mathcal{X}_j\} = \emptyset, \forall\, i \neq j. \tag{9}$$

Pairwise Convex-Separability of Multiple Categories: We say m pattern sets $\{\mathcal{X}_i\}_{i=1}^m$ are pairwise convexly-separable if every two pattern sets are mutually convexly-separable, that is,

$$\begin{aligned} \mathrm{CH}\{\mathcal{X}_i\} &\cap \mathcal{X}_j = \emptyset, \forall\, i \neq j \\ \mathrm{CH}\{\mathcal{X}_j\} &\cap \mathcal{X}_i = \emptyset, \forall\, i \neq j. \end{aligned} \tag{10}$$

Linear-Separability of Multiple Categories: We say m pattern sets $\{\mathcal{X}_i\}_{i=1}^m$ are linearly-separable if every pattern set $\mathcal{X}_i$ is linearly-separable from the union of all the other pattern sets, that is,

$$\mathrm{CH}\{\mathcal{X}_i\} \cap \mathrm{CH}\left\{\cup_{j\neq i}\mathcal{X}_j\right\} = \emptyset. \tag{11}$$

According to this definition, there exists an m-dimensional linear classifier such that each pattern is positive in one axis and all of the other patterns are negative in this axis.

For multiple category classification, linear separability is much stronger than pairwise linear separability.

3.1.2. Separability of Multiple Pattern Sets with ReNN

Given m pattern sets, namely $\mathcal{X}_i, i \in [m]$, we call them separable by ReNNs, which may have additional sign constraints, if an ReNN (with corresponding constraints if any) exists such that

$$y_i(\mathbf{x}) \begin{cases} > 0, & \forall\, \mathbf{x} \in \mathcal{X}_i \\ < 0, & \forall \mathbf{x} \in \bigcup_{j\neq i}\mathcal{X}_j \end{cases} \tag{12}$$

hold for all $i \in [m]$.

3.2. Binary Classification Capacity of Single Hidden Layer Networks

For binary classification, we only need one dimensional classifiers, and the sign-constrained ReNN defined in Eq. (3) can be described as

$$f(\mathbf{x}) \triangleq \mathbf{a}^T \max(\mathbf{0}, W^T\mathbf{x} + \mathbf{b}) + c \tag{13}$$

where the output layer weight matrix A is reduced to a vector $\mathbf{a} \in \mathbb{R}^m$ and the bias vector $\mathbf{c}$ is reduced to be a scalar c.

Next, we establish the connections between sign constrained rectifier networks and convexly-separable pattern sets. For pattern sets $\mathcal{X}_+$ and $\mathcal{X}_-$ labelled positive and negative respectively, a single-hidden-layer binary classifier $f(\mathbf{x})$, as defined in Eq. (13), is called a single hidden layer separator of $\mathcal{X}_+$ and $\mathcal{X}_-$ if it satisfies

$$\begin{array}{rcl} f(\mathbf{x}) & > & 0,\ \forall\, \mathbf{x} \in \mathcal{X}_+ \\ f(\mathbf{x}) & < & 0,\ \forall\, \mathbf{x} \in \mathcal{X}_-. \end{array} \tag{14}$$

If it further satisfies $\mathbf{a} \preceq 0, c \geq 0$, we call it a *sign-constrained single-hidden-layer separator* of $\mathcal{X}_+$ and $\mathcal{X}_-$.

Lemma 4. *Let $\mathcal{X}_+, \mathcal{X}_-$ be a pair of finite pattern sets in $\mathbb{R}^n$ and be labelled positive and negative respectively. Then $\mathcal{X}_+, \mathcal{X}_-$ can be separated by a sign-constrained single-hidden-layer classifier, as defined in Eq. (13) and satisfying $c \geq 0$ and $\mathbf{a} \preceq 0$, if and only if the positive pattern set $\mathcal{X}_+$ is convexly-separable from the negative pattern set $\mathcal{X}_-$, i.e., $\mathcal{X}_- \cap \mathrm{CH}(\mathcal{X}_+) = \emptyset$.*

Proof: (Sufficiency). Suppose $\mathrm{CH}(\mathcal{X}_+) \cap \mathcal{X}_- = \emptyset$. Let n_- be the number of training patterns in $\mathcal{X}_-$ and $\mathbf{x}_i^-$ be the i^{th} member of $\mathcal{X}_-$. Since $\mathbf{x}_i^- \notin \mathrm{CH}(\mathcal{X}_+)$ for any $i \in [n_-]$, there exists $\mathbf{w}_i, b_i, i \in [n_-]$ such that

$$\begin{array}{rcl} \mathbf{w}_i^T\mathbf{x}_i^- + b_i & > & 0 \\ \mathbf{w}_i^T\mathbf{x} + b_i & < & 0,\ \forall\, \mathbf{x} \in \mathcal{X}_+. \end{array} \tag{15}$$

Note that the responses of positive patterns are negative, making $\mathcal{X}_+$ shrinking into a single point $\mathbf{0}$.

Denote

$$\begin{array}{rcl} W & = & [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_{n_-}] \\ \mathbf{b} & = & [b_1, b_2, \cdots, b_{n_-}]^T \\ \mathbf{z} & = & \max(\mathbf{0}, W^T\mathbf{x} + \mathbf{b}). \end{array} \tag{16}$$

Then we have

$$\begin{array}{rcl} \mathcal{Z}_+ & \triangleq & \{\mathbf{z} = \max(\mathbf{0}, W^T\mathbf{x} + \mathbf{b}) : \mathbf{x} \in \mathcal{X}_+\} \\ & = & \{\mathbf{0}\} \\ \mathcal{Z}_- & \triangleq & \{\mathbf{z} = \max(\mathbf{0}, W^T\mathbf{x} + \mathbf{b}) : \mathbf{x} \in \mathcal{X}_-\} \\ & \subset & \{\mathbf{z} : \mathbf{1}^T\mathbf{z} > \gamma_{\min}, \mathbf{z} \neq \mathbf{0},\ z_i \geq 0, \forall\, i \in [n_-].\} \end{array} \tag{17}$$

where

$$\begin{array}{rcl} \gamma_{\min} & \triangleq & \min\limits_{\mathbf{x}\in\mathcal{X}_-} \mathbf{1}^T \max(\mathbf{0}, W^T\mathbf{x} + \mathbf{b}) \\ & > & 0. \end{array} \tag{18}$$

For a single-hidden-layer binary classifier $f(\mathbf{x})$, as described in Eq. (13), if we choose $c = 1$ and $\mathbf{a} = -\frac{2}{\gamma_{\min}}\mathbf{1} \preceq 0$, then

$$f(\mathbf{x}) = -\frac{2}{\gamma_{\min}}\mathbf{1}^T \max(0, W^T\mathbf{x} + \mathbf{b}) + 1$$

satisfies

$$\begin{array}{rcccl} f(\mathbf{x}) & \leq & -1 & < & 0,\ \forall\, \mathbf{x} \in \mathcal{X}_-, \\ f(\mathbf{x}) & = & 1 & > & 0,\ \forall\, \mathbf{x} \in \mathcal{X}_+ \end{array} \tag{19}$$

which implies that $\mathcal{X}_+$ and $\mathcal{X}_-$ can be separated by a sign-constrained single-hidden-layer binary classifier.

(Necessity). Suppose that $\mathcal{X}_+, \mathcal{X}_-$ can be separated by a sign-constrained single-hidden-layer binary classifier with $\mathbf{a} \preceq 0, c \geq 0$ such that $f(\mathbf{x})$, as defined in Eq. (13), satisfies Eq. (14). Next, we will prove the convexity of the set $\{\mathbf{x} : f(\mathbf{x}) > 0\}$ and show that $f(\mathbf{x}) > 0$ holds for all $\mathbf{x}$ in the convex hull of $\mathcal{X}_+$.

Let z_0, z_1 be two arbitrary real numbers and let $z_\lambda = \lambda z_1 + (1-\lambda)z_0$ be their linear combination. Since

$$\max(0, z_\lambda) \leq \lambda \max(0, z_1) + (1-\lambda)\max(0, z_0),\ \forall\, \lambda \in [0, 1], \tag{20}$$

we have

$$\mathbf{a}^T \max(0, \mathbf{z}_\lambda) \geq \lambda \mathbf{a}^T \max(0, \mathbf{z}_0) + (1-\lambda)\mathbf{a}^T \max(0, \mathbf{z}_1), \quad \forall\, \lambda \in [0, 1] \tag{21}$$

for any $\mathbf{a} \preceq 0$ and any $\mathbf{z}_0, \mathbf{z}_1$ with the same dimensions, where $\mathbf{z}_\lambda \triangleq \lambda\mathbf{z}_1 + (1-\lambda)\mathbf{z}_0$.

In particular, let $\mathbf{z}_\lambda = W^T\mathbf{x}_\lambda + \mathbf{b}$ with $\mathbf{x}_\lambda = \lambda\mathbf{x}_1 + (1-\lambda)\mathbf{x}_0$. Then we have

$$\begin{array}{rcl} f(\mathbf{x}_\lambda) & = & \mathbf{a}^T \max(0, \mathbf{z}_\lambda) + \beta \\ & \geq & \lambda\left[\mathbf{a}^T \max(0, \mathbf{z}_0) + \beta\right] + \\ & & (1-\lambda)\left[\mathbf{a}^T \max(0, \mathbf{z}_1) + \beta\right] \\ & = & \lambda f(\mathbf{x}_0) + (1-\lambda) f(\mathbf{x}_1), \forall\, \lambda \in [0, 1] \end{array} \tag{22}$$

and therefore

$$f(\mathbf{x}_\lambda) > 0,\ \forall\, \lambda \in [0, 1] \tag{23}$$

if and only if

$$f(\mathbf{x}_\lambda) > 0,\ \forall\ \lambda = 0, 1. \tag{24}$$

Hence $\{\mathbf{x} : f(\mathbf{x}) > 0\}$ is a convex set, and thus

$$f(\mathbf{x}) > 0, \forall\ \mathbf{x} \in \mathrm{CH}(\mathcal{X}_+) \tag{25}$$

follows from $f(\mathbf{x}) > 0, \forall\ \mathbf{x} \in \mathcal{X}_+$. Note that $f(\mathbf{x}) < 0$ for all $\mathbf{x} \in \mathcal{X}_-$ (from Eq. (14)). So $\mathcal{X}_-$ and $\mathrm{CH}(\mathcal{X}_+)$ are separable and thus $\mathrm{CH}(\mathcal{X}_+) \cap \mathcal{X}_- = \emptyset$, which completes the proof.

□

3.3. Data Decomposition

The following Lemma shows the capacity of sign-constrained single-hidden-layer classifiers in decomposing the negative pattern set into several subsets so that each subset is linearly separable from the positive pattern set.

Lemma 5. *Let $\mathcal{X}_+$ be a pattern set which is convexly-separable from $\mathcal{X}_-$, and let $f(\mathbf{x})$, as defined in Eq.(13) with l hidden nodes and satisfying $\mathbf{a} \preceq 0, c \geq 0$, be one of their sign-constrained single-hidden-layer separators. for any subset of $[l] \triangleq \{1, 2, \cdots, l\}$, namely $\mathcal{I} \subset [l]$, define*

$$f_{\mathcal{I}}(\mathbf{x}) \triangleq \left(\sum_{i\in\mathcal{I}} a_i(\mathbf{w}_i^T\mathbf{x} + b_i)\right) + c \tag{26}$$

and

$$\mathcal{X}_-^{\mathcal{I}} \triangleq \{\mathbf{x} : f_{\mathcal{I}}(\mathbf{x}) < 0, \mathbf{x} \in \mathcal{X}_-\}. \tag{27}$$

Then we have

$$X_- = \bigcup_{\mathcal{I}\subset[l]} \mathcal{X}_-^{\mathcal{I}} \tag{28}$$

and

$$\mathrm{CH}\left(\mathcal{X}_-^{\mathcal{I}}\right) \cap \mathrm{CH}(\mathcal{X}_+) = \emptyset, \tag{29}$$

i.e, $\mathcal{X}_-^{\mathcal{I}}$ and $\mathcal{X}_+$ are linearly-separable, and furthermore, $f_{\mathcal{I}}(\mathbf{x})$ is their linear separator satisfying

$$\begin{array}{rcl} f_{\mathcal{I}}(\mathbf{x}) & < & 0,\ \forall\ \mathbf{x} \in \mathcal{X}_-^{\mathcal{I}} \\ f_{\mathcal{I}}(\mathbf{x}) & > & 0,\ \forall\ \mathbf{x} \in \mathcal{X}_+. \end{array} \tag{30}$$

Proof: From $\mathbf{a} \preceq 0$, it follows that

$$f_{\mathcal{I}}(\mathbf{x}) \geq f(\mathbf{x}),\ \forall\ \mathcal{I} \subset [l], \mathbf{x} \in \mathbb{R}^n \tag{31}$$

and consequently

$$f_{\mathcal{I}}(\mathbf{x}) \quad > \quad 0,\ \forall\ \mathcal{I} \subset [l], \mathbf{x} \in \mathcal{X}_+. \tag{32}$$

Then Eq. (30) follows directly from Eq. (32) and the definition of $\mathcal{X}_-^{\mathcal{I}}$ in Eq. (27). Note that $f_{\mathcal{I}}(\mathbf{x})$ is a linear classifier satisfying Eq. (30), $f_{\mathcal{I}}(\mathbf{x})$ is a linear separator of $\mathcal{X}_-^{\mathcal{I}}$ and $\mathcal{X}_+$, and Eq. (29) holds consequently.

To complete the proof, it remains to prove Eq. (28). Let $\mathbf{x} \in \mathcal{X}_-$ be any pattern with negative label and let $\mathcal{I} \subset [l]$ be the index set so that $\mathbf{w}_i^T\mathbf{x} + b_i > 0$ for all $i \in \mathcal{I}$ and $\mathbf{w}_i^T\mathbf{x} + b_i \leq 0$ for all $i \notin \mathcal{I}$. Then $f_{\mathcal{I}}(\mathbf{x}) = f(\mathbf{x}) < 0$ and thus $\mathbf{x} \in \mathcal{X}_-^{\mathcal{I}}$. This proves that any element in $\mathcal{X}_-$ is in $\mathcal{X}_-^{\mathcal{I}}$ for some $\mathcal{I} \subset [l]$. Hence Eq. (28) is true and the proof is completed.

□

3.3.1. Disjoint Subset Decomposition

There are $(2^l - 1)$ non-empty subsets in $[l] \triangleq \{1, 2, \cdots, l\}$ and therefore $\mathcal{X}_-$ can be decomposed into $(2^l - 1)$ subsets using Eq. (28) in Lemma 5. However, if the number of patterns in $\mathcal{X}_-$ is smaller than $(2^l - 1)$, many of the subsets $\mathcal{X}_-^{\mathcal{I}}$ in Eq. (28) are empty or redundant. Next, we present an efficient algorithm to decompose $\mathcal{X}_-$ into a number of disjoint subsets, wherein each subset is linearly separable from $\mathcal{X}_+$.

Let $\mathbf{x}$ be a given example in the dataset $\mathcal{X}_-$ and let

$$\mathcal{I}(\mathbf{x}) \triangleq \{i : \mathbf{w}_i^T\mathbf{x} + b_i > 0, i \in [l]\} \tag{33}$$

denote the set of nodes where the responses of $\mathbf{x}$ are positive. Then we have

$$f(\mathbf{x}) = f_{\mathcal{I}(\mathbf{x})}(\mathbf{x}). \tag{34}$$

Based on this relationship, the first step of the proposed decomposition algorithm is to search for $\mathcal{I}_1 \in \{I(\mathbf{x}) : \mathbf{x} \in \mathcal{X}_-\}$ so that

$$\Omega_1 \triangleq \{\mathbf{x} \in \mathcal{X}_- : f_{\mathcal{I}_1}(\mathbf{x}) < 0\} \tag{35}$$

has the largest number of patterns. In the second step, we discard the patterns in Ω_1 and search for

$$\mathcal{I}_2 \in \{I(\mathbf{x}) : \mathbf{x} \in \mathcal{X}_-\backslash\Omega_1\} \tag{36}$$

so that the number of patterns in

$$\Omega_2 \triangleq \{\mathbf{x} \in \mathcal{X}_-\backslash\Omega_1 : f_{\mathcal{I}_j}(\mathbf{x}) < 0, j = 1, 2\} \tag{37}$$

is the largest. Similarly, in the k^{th} step, we discard all the patterns in Ω_j $(j = 1, 2, \cdots, k-1)$, and search for

$$\mathcal{I}_k \in \left\{ I(\mathbf{x}) : \mathbf{x} \in \mathcal{X}_-\backslash\bigcup_{j=1}^{k-1}\Omega_j \right\} \tag{38}$$

to maximize the number of patterns in

$$\Omega_k \triangleq \left\{ \mathbf{x} \in \mathcal{X}_-\backslash\bigcup_{j=1}^{k-1}\Omega_j : f_{\mathcal{I}_j}(\mathbf{x}) < 0, 1 \leq j \leq k \right\}. \tag{39}$$

Since the sets Ω_j $(j = 1, 2, \cdots, k)$ are disjoint and non-empty, this procedure converges and there exists an integer K such that

$$\mathcal{X}_- = \bigcup_{j=1}^{K}\Omega_j. \tag{40}$$

Let

$$\hat{f}(\mathbf{x}) = \min_{j=1}^{K} f_{\mathcal{I}_j}(\mathbf{x}), \tag{41}$$

then we have, $\hat{f}(\mathbf{x}) > 0$ for any $\mathbf{x} \in \mathcal{X}_+$ and $\hat{f}(\mathbf{x}) < 0$ for any $\mathbf{x} \in \mathcal{X}_-$. Hence, $\hat{f}(\mathbf{x})$ is a separator of the pattern sets and is a good approximation of the original convex classifier $f(\mathbf{x})$. In Section 6, an example is provided to show that a learnt model, with $l = 500$, can be well approximated by the minimum of 40 (i.e., $K = 40$) linear classifiers, while the original classifier is the minimum of $2^{500} - 1 \approx 3 \times 10^{150}$ linear classifiers. This decomposition and approximation are useful in analysing the key discrimination factors of the learnt neural network classifiers.

3.4. Multiple Category Classification with Single Hidden Layers

This section considers multiple category classification problems. We will show that multiple (equal to or more than three) sets can be transformed to be linearly separable by a single hidden layer *if and only if* every pair of the classes are mutually convexly-separable.

For classification of m classes, an m-dimensional classifier, namely $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_m(\mathbf{x})]$, with sign-constrained ReNN can be described as

$$f_i(\mathbf{x}) = \mathbf{a}_i^T \max(0, W^T\mathbf{x} + \mathbf{b}) + c_i, \mathbf{a}_i \preceq 0, i = 1, 2, \cdots, m. \tag{42}$$

Theorem 6. *Multiple pattern sets, namely $\mathcal{X}_i (i = 1, 2, \cdots, m)$ can be separated by a sign-constrained single-hidden-layer classifier, as defined in Eq. (42), if and only if these pattern sets are pairwise mutually convex-separable, i.e., $\mathrm{CH}(\mathcal{X}_i) \cap \mathcal{X}_j = \emptyset$ for any $i \neq j$.*

Proof: From $\mathrm{CH}(\mathcal{X}_i) \cap \mathcal{X}_j = \emptyset, \ \forall\, j \neq i$, it follows that

$$\mathrm{CH}(\mathcal{X}_i) \cap \{\cup_{j\neq i} \mathcal{X}_j\} = \emptyset, \ \forall\, i \in [m] \tag{43}$$

which implies that each pattern set is convexly-separable from the union of all the other patterns.

Let $\mathcal{X}_+ = \mathcal{X}_i$ and $\mathcal{X}_- = \bigcup_{j\neq i} \mathcal{X}_j$. By *Lemma* 4, there is a sign-constrained single-hidden-layer classifier, namely

$$f_i(\mathbf{x}) \triangleq \mathbf{a}_i^T \max\{\mathbf{0}, W_i^T\mathbf{x} + \mathbf{b}_i\} + c_i \tag{44}$$

with $\mathbf{a}_i \preceq 0$ and $c_i \geq 0$, such that

$$\begin{array}{ll} f_i(\mathbf{x}) > 0, & \forall \mathbf{x} \in \mathcal{X}_i \\ f_i(\mathbf{x}) < 0, & \forall \mathbf{x} \in \bigcup_{j\neq i} \mathcal{X}_j. \end{array} \tag{45}$$

Let the multiple output SCReNN1 in Eq. (3) be defined with

$$\begin{array}{rcl} A & = & \begin{bmatrix} \mathbf{a}_1 & & \\ & \ddots & \\ & & \mathbf{a}_m \end{bmatrix} \\ W & = & [W_1, W_2, \cdots, W_m] \\ \mathbf{b} & = & \left[\mathbf{b}_1^T, \mathbf{b}_2^T, \cdots, \mathbf{b}_m^T\right]^T \\ \mathbf{c} & = & [c_1, c_2, \cdots, c_m]^T. \end{array} \tag{46}$$

Then the output at the i^{th} node, namely y_i, equals to $f_i(\mathbf{x})$, and therefore

$$\begin{array}{ll} y_i > 0, & \forall\, \mathbf{x} \in \mathcal{X}_i \\ y_i < 0, & \forall\, \mathbf{x} \in \bigcup_{j\neq i} \mathcal{X}_j \end{array} \tag{47}$$

hold for all $i \in [m]$. This proves the sufficiency of mutual convex-separability for multiple category pattern sets to be separable by a single-hidden-layer SCReNN. Next, we prove its necessity. Suppose a single-hidden-layer SCReNN exists to separate m category pattern sets $\{\mathcal{X}_i\}_{i=1}^m$ such that

$$y_i(\mathbf{x}) \triangleq \mathbf{a}_i^T \max\{\mathbf{0}, W^T\mathbf{x} + \mathbf{b}\} + c_i \tag{48}$$

satisfies: $y_i(\mathbf{x}) > 0$ for all $\mathbf{x} \in \mathcal{X}_i$ and $y_i(\mathbf{x}) < 0$ for all $\mathbf{x} \in \bigcup_{j\neq i} \mathcal{X}_j$. Note that $\mathbf{a}_i \preceq 0$, $y_i(\mathbf{x})$ is a sign constrained single-hidden-layer separator of $\mathcal{X}_i$ from the set $\bigcup_{j\neq i} \mathcal{X}_j$. Then by Lemma 4, $\mathcal{X}_i$ is convexly-separable from the union of all the other pattern sets, and therefore $\mathcal{X}_i$ is convexly-separable from any other pattern set $\mathcal{X}_j (j \neq i)$. Note that, this is true for all $i \in [m]$ and therefore every pair of the pattern sets are mutually convexly-separable, which completes the proof for the necessity of pairwise mutual convex-separability.

□

Note that each element of the m dimensional multiple category classifier is a sign-constrained ReNN to separate one pattern set from the others, the decomposition property of such sign-constrained multiple category classifiers can be derived directly from Lemma 5.

4. The Capacity of Sign-Constrained Rectifier Networks with Two Hidden Layers

In this section, we first investigate the universal classification power of sign-constrained two-hidden-layer binary classifiers and their capacity to decompose one pattern set into smaller subsets so that each subset is convexly separable from the other pattern set. We then extend this result to multiple category classification problems.

4.1. Binary Classification with Two Hidden Layers

A two-hidden-layer binary classifier, with n dimensional input, l_1 bottom hidden nodes, l_2 top hidden nodes and a single output, can be described by

$$\begin{array}{rcl} f\{\mathbf{g}(\mathbf{x})\} & = & \mathbf{a}^T \max(\mathbf{0}, W_2^T\mathbf{g}(\mathbf{x}) + \mathbf{b}_2) + c \\ \mathbf{g}(\mathbf{x}) & = & \max(\mathbf{0}, W_1^T\mathbf{x} + \mathbf{b}_1) \end{array} \tag{49}$$

where c is a scalar number, $\mathbf{a} \in \mathbb{R}^{l_2}, \mathbf{b} \in \mathbb{R}^{l_2}, \mathbf{b}_1 \in \mathbb{R}^{l_1}, W_2 \in \mathbb{R}^{l_1 \times l_2}$ and $W_1 \in \mathbb{R}^{n \times l_1}$.

We say that a two-hidden-layer binary classifier $f\{\mathbf{g}(\mathbf{x})\}$, as defined in Eq. (49), is a two-hidden-layer separator of $\mathcal{X}_+$ and $\mathcal{X}_-$ if it satisfies

$$\begin{array}{rcl} f\{\mathbf{g}(\mathbf{x})\} & > & 0, \ \forall\, \mathbf{x} \in \mathcal{X}_+ \\ f\{\mathbf{g}(\mathbf{x})\} & < & 0, \ \forall\, \mathbf{x} \in \mathcal{X}_-. \end{array} \tag{50}$$

If it further satisfies $\mathbf{a} \preceq 0$ and $W_2 \preceq 0$, we call it a *sign-constrained two-hidden-layer separator* of $\mathcal{X}_+$ and $\mathcal{X}_-$.

Lemma 7. *For any two disjoint pattern sets, namely $\mathcal{X}_+$ and $\mathcal{X}_-$, in $\mathbb{R}^n$, there exists a sign-constrained two-hidden-layer binary classifier $f\{\mathbf{g}(\mathbf{x})\}$, as defined in Eq. (49) and satisfying $\mathbf{a} \preceq 0, c \geq 0, W_2 \preceq 0, \mathbf{b}_2 \succeq 0$, such that $f\{\mathbf{g}(\mathbf{x})\} > 0$ for all $\mathbf{x} \in \mathcal{X}_+$ and $f\{\mathbf{g}(\mathbf{x})\} < 0$ for all $\mathbf{x} \in \mathcal{X}_-$.*

Proof: Let

$$\mathcal{X}_+ = \bigcup_{i=1}^{L_1} \mathcal{X}_+^i, \mathcal{X}_- = \bigcup_{j=1}^{L_2} \mathcal{X}_-^j \tag{51}$$

be the disjoint convex hull decomposition [33] of $\mathcal{X}_+$ and $\mathcal{X}_-$. Then we have

$$\mathrm{CH}(\mathcal{X}_+^i) \cap \mathrm{CH}(\mathcal{X}_-^j) \neq \emptyset, \ \forall\, i \in [L_1], j \in [L_2] \tag{52}$$

which implies that

$$\mathcal{X}_+ \cap \mathrm{CH}(\mathcal{X}_-^i) \neq \emptyset,\ \forall\, i \in [L_2]. \tag{53}$$

Apply Lemma 4 on $\mathcal{X}_-^i$ and $\mathcal{X}_+$ where $\mathcal{X}_-^i$ is treated as the positive pattern set and $\mathcal{X}_+$ is treated as the negative pattern set. Then, for each i, there exists a sign-constrained single-hidden-layer separator between $\mathcal{X}_+$ and $\mathcal{X}_-^i$. More precisely, there exist $\mathbf{w}_i \preceq 0, b_i \geq 0, W_1, \mathbf{b}_1$ such that

$$\begin{array}{rcl} g_i(\mathbf{x}) & < & 0,\ \forall \mathbf{x} \in \mathcal{X}_+ \\ g_i(\mathbf{x}) & > & 0,\ \forall \mathbf{x} \in \mathcal{X}_-^i \end{array} \tag{54}$$

where

$$g_i(\mathbf{x}) \triangleq \mathbf{w}_i^T \max(0, W_1^T\mathbf{x} + \mathbf{b}_1) + b_i. \tag{55}$$

Let $W_2 = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_{L_2}] \preceq 0$, $\mathbf{b}_2 = [b_1, b_2, \cdots, b_{L_2}]^T \succeq 0$ and consider the transformation

$$\mathbf{z} = \mathbf{g}(\mathbf{x}) \triangleq \max(0, W_2^T \max(0, W_1^T\mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2). \tag{56}$$

Denote

$$\begin{array}{rcl} \mathcal{Z}_+ & \triangleq & \{\mathbf{z} : \mathbf{z} = \mathbf{g}(\mathbf{x}), \mathbf{x} \in \mathcal{X}_+\} \\ & = & \{\mathbf{0}\} \\ \mathcal{Z}_- & \triangleq & \{\mathbf{z} : \mathbf{z} = \mathbf{g}(\mathbf{x}), \mathbf{x} \in \mathcal{X}_-\} \\ & \subset & \{\mathbf{z} : \mathbf{1}^T\mathbf{z} > \gamma_{\min}, \mathbf{z} \neq 0, z_i \geq 0, \forall i \in [L_1]\} \end{array} \tag{57}$$

where

$$\begin{array}{rcl} \gamma_{\min} & \triangleq & \min\limits_{\mathbf{x} \in \mathcal{X}_-} \mathbf{1}^T \max(0, W_2^T\mathbf{g}(\mathbf{x}) - \mathbf{b}_2) \\ & > & 0. \end{array} \tag{58}$$

Let $\mathbf{a} = -\frac{2}{\gamma_{\min}}\mathbf{1} \succeq 0, c = 1$ and $f(\mathbf{z}) \triangleq \mathbf{a}^T\mathbf{z} + c$. Then $f(\mathbf{z}) \leq -1$ for any $\mathbf{z} \in \mathcal{Z}_-$ and $f(\mathbf{z}) = 1$ for any $\mathbf{z} \in \mathcal{Z}_+$. Hence

$$f\{\mathbf{g}(\mathbf{x})\} \triangleq \mathbf{a}^T \max(0, W_2^T\mathbf{g}(\mathbf{x}) + \mathbf{b}_2) + c \tag{59}$$

satisfies: $f\{\mathbf{g}(\mathbf{x})\} > 0$ for $\mathbf{x} \in \mathcal{X}_+$ and $f\{\mathbf{g}(\mathbf{x})\} < 0$ for $\mathbf{x} \in \mathcal{X}_-$. Note that $W_2 \preceq 0, \mathbf{b}_2 \succeq 0, \mathbf{a} \preceq 0, c > 0$, $f\{\mathbf{g}(\mathbf{x})\}$ is a sign-constrained two-hidden-layer binary classifier, and the proof is completed.

□

4.2. Data Decomposition with Two Hidden Layers

Next, we investigate the applications of the two-hidden-layer sign constrained ReNN classifier to decompose one pattern set (labelled positive) into several subsets so that each subset is convexly separable from the other pattern set.

Lemma 8. *Let $\mathcal{X}_+, \mathcal{X}_-$ be two disjoint pattern sets and let $f\{\mathbf{g}(\mathbf{x})\}$, as defined in Eq. (49) and satisfying $\mathbf{a} \preceq 0, c \geq 0, W_2 \preceq 0, \mathbf{b}_2 \succeq 0$, be one of their sign-constrained two-hidden-layer binary separators with l_2 top hidden nodes and satisfying Eq. (50). Let $\mathbf{w}_i$ denote the i^{th} column of W_2, b_i denote the i^{th} element of $\mathbf{b}_2$, and define*

$$\begin{array}{rcl} f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\} & \triangleq & \left(\sum\limits_{i \in \mathcal{I}} a_i[\mathbf{w}_i^T\mathbf{g}(\mathbf{x}) + b_i]\right) + c \\ \mathcal{X}_-^{\mathcal{I}} & \triangleq & \{\mathbf{x} : f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\} < 0, \mathbf{x} \in \mathcal{X}_-\} \end{array} \tag{60}$$

for any subset, namely $\mathcal{I}$, in $[l_2]$. Then we have

$$X_- = \bigcup_{\mathcal{I} \subset [m]} \mathcal{X}_-^{\mathcal{I}} \tag{61}$$

and

$$\mathrm{CH}\left(\mathcal{X}_-^{\mathcal{I}}\right) \cap \mathcal{X}_+ = \emptyset, \tag{62}$$

i.e, $\mathcal{X}_-^{\mathcal{I}}$ is convexly-separable from $\mathcal{X}_+$. Furthermore, $f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\}$ is their single-hidden-layer separator satisfying

$$\begin{array}{rcl} f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\} & > & 0,\ \forall\, \mathbf{x} \in \mathcal{X}_+ \\ f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\} & < & 0,\ \forall\, \mathbf{x} \in \mathcal{X}_-^{\mathcal{I}}. \end{array} \tag{63}$$

Proof: Note that $a_i \leq 0$, we have

$$f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\} \geq f\{\mathbf{g}(\mathbf{x})\},\ \forall\, \mathbf{x} \in \mathbb{R}^n \tag{64}$$

and therefore

$$f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\} > 0,\ \forall\, \mathbf{x} \in \mathcal{X}_+ \tag{65}$$

which proves the first inequality of Eq. (63) while the second one follows from the definition of $\mathcal{X}_-^{\mathcal{I}}$ in Eq. (60). Hence, $f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\}$ is a single-hidden-layer separator of $\mathcal{X}_+$ and $\mathcal{X}_-^{\mathcal{I}}$. Note that

$$\begin{array}{rcl} f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\} & = & \mathbf{a}_{\mathcal{I}}^T\mathbf{g}(\mathbf{x}) + c_{\mathcal{I}} \\ & = & \mathbf{a}_{\mathcal{I}}^T \max(0, W_1^T\mathbf{x} + \mathbf{b}_1) + c_{\mathcal{I}} \end{array} \tag{66}$$

where

$$\begin{array}{rcl} \mathbf{a}_{\mathcal{I}} & = & \sum\limits_{i \in \mathcal{I}} a_i\mathbf{w}_i \succeq 0 \\ c_{\mathcal{I}} & = & \sum\limits_{i \in \mathcal{I}} a_i b_i + c. \end{array} \tag{67}$$

which imply that $(-f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\})$ is a *sign-constrained* single-hidden-layer separator of $\mathcal{X}_-^{\mathcal{I}}$ from $\mathcal{X}_+$. Then by Lemma 4, we have Eq. (62).

Now it remains to prove Eq. (61). It suffices to prove that, for any $\mathbf{x} \in \mathcal{X}_-$, there exists $\mathcal{I} \subset [l_2]$ such that $\mathbf{x} \in \mathcal{X}_-^{\mathcal{I}}$. Let $\mathbf{x}$ be a member in $\mathcal{X}_-$ and let $\mathcal{I} \subset [l_2], \mathcal{I} \neq \emptyset$ be the index set such that $\mathbf{w}_i^T\mathbf{g}(\mathbf{x}) + b_i > 0$ for all $i \in \mathcal{I}$ and $\mathbf{w}_i^T\mathbf{g}(\mathbf{x}) + b_i \leq 0$ for all $i \notin \mathcal{I}$. Then $f_{\mathcal{I}}\{\mathbf{g}(\mathbf{x})\} = f\{\mathbf{g}(\mathbf{x})\} < 0$ and thus $\mathbf{x}$ is in $\mathcal{X}_-^{\mathcal{I}}$.

□

Lemma 8 states that the negative pattern set can be decomposed into several subsets by a two-hidden-layer SCReNN, namely, $\mathcal{X}_- = \bigcup\limits_{i=1}^{t} \mathcal{X}_-^i$, so that each $\mathcal{X}_-^i$ is convexly-separable from $\mathcal{X}_+$. Then by labelling $\mathcal{X}_-^i$ as positive and $\mathcal{X}_+$ as negative, and from Lemma 4, $\mathcal{X}_+$ can be decomposed into a number, namely t_i, of subsets by a single-hidden-layer SCReNN, namely, $\mathcal{X}_+ = \bigcup\limits_{j=1}^{t_i} \mathcal{X}_+^j$, so that $\mathcal{X}_+^i$ and $\mathcal{X}_-^j$ are linearly-separable. Hence, one can investigate the discriminant features

of the two patterns by using the linear classifiers of these subsets of the patterns. With the decomposed subsets, one can investigate the pattern structures. The numbers of the subsets are determined by the numbers of hidden nodes in the top hidden layers of the two-hidden-layer SCReNNs and the numbers of the hidden nodes of the single-hidden-layer SCReNNs. To find compact pattern structures and meaningful discriminant features, one may apply the efficient decomposition algorithm presented in Sec 3.3.1.

4.3. Multiple Category Classification

This section extends the results of the last section to multiple category classification problems. An m-dimensional classifier with two-hidden-layer SCReNN, namely $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_m(\mathbf{x})]$, can be described as

$$\begin{array}{rcl} f_k(\mathbf{x}) & = & \mathbf{a}_i^T \max(0, W_2^T \max\{0, W_1^T\mathbf{x} + \mathbf{b}_1\} + \mathbf{b}_2) + \mathbf{c} \\ \mathbf{a}_k & \preceq & 0, W_2 \preceq 0, k = 1, 2, \cdots, m. \end{array} \tag{68}$$

Theorem 9. *Let $\{\mathcal{X}_k, k = 1, 2, \cdots, m\}$ be m disjoint pattern sets with a finite number of points, then there exists an m-dimensional classifier with two-hidden-layer* SCReNN*, as defined in Eq. (68), such that for each $k = 1, 2, \cdots, m, f_k(\mathbf{x})$ is positive for any $\mathbf{x} \in \mathcal{X}_k$, and negative for any $\mathbf{x}$ from other pattern sets.*

Proof: Denote $\hat{\mathcal{X}}_k \triangleq \bigcup_{l=1, l\neq k}^m \mathcal{X}_l$. So $\mathcal{X}_k$ and $\hat{\mathcal{X}}_k$ are disjoint. From Lemma 7, there exists two hidden layer SCReNNs

$$\begin{array}{rcl} f_k\{G_k(\mathbf{x})\} & = & \bar{\mathbf{a}}_k^T \max(\mathbf{0}, U_k^T G_k(\mathbf{x}) + \mathbf{b}_{1,k}) + c_k \\ G_k(\mathbf{x}) & = & \max(\mathbf{0}, V_k^T\mathbf{x} + \mathbf{b}_{2,k}) \\ \bar{\mathbf{a}} \preceq 0, & & U_k \preceq 0 \end{array} \tag{69}$$

such that

$$\begin{array}{rcl} f_k\{G_k(\mathbf{x})\} & > & 0, \ \forall\, \mathbf{x} \in \mathcal{X}_k \\ f_k\{G_k(\mathbf{x})\} & < & 0, \ \forall\, \mathbf{x} \in \hat{\mathcal{X}}_k. \end{array} \tag{70}$$

Denote

$$\begin{array}{rcl} W_1 & \triangleq & [V_1, V_2, \cdots, V_m] \\ \mathbf{b}_1 & \triangleq & [\mathbf{b}_{1,1}^T, \mathbf{b}_{1,2}^T, \cdots, \mathbf{b}_{1,m}^T]^T \\ \mathbf{b}_2 & \triangleq & [\mathbf{b}_{2,1}^T, \mathbf{b}_{2,2}^T, \cdots, \mathbf{b}_{2,m}^T]^T \\ \mathbf{c} & \triangleq & [c_1, c_2, \cdots, c_m]^2 \end{array} \tag{71}$$

and

$$\begin{array}{rcl} W_2 & = & \begin{bmatrix} U_1 & 0 & \cdots & 0 \\ 0 & U_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & U_m \end{bmatrix} \\ A & = & \begin{bmatrix} \bar{\mathbf{a}}_1 & 0 & \cdots & 0 \\ 0 & \bar{\mathbf{a}}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \bar{\mathbf{a}}_m \end{bmatrix}. \end{array} \tag{72}$$

Now let the weights and bias of the two-hidden-layer ReNN, defined in Eq. (68), be chosen as in Eq. (71) and Eq. (72), then the output $\mathbf{y}$ of the network satisfies $y_k = f_k\{G_k(\mathbf{x})\}$. Note that $A \preceq 0, W_2 \preceq 0$. The defined ReNN is a two-hidden-layer sign-constrained ReNN and this completes the proof of the universal classification power of two-hidden-layer SCReNN.

□

Next we address the decomposition capacity of two-hidden-layer rectifier neural networks. Suppose a two-hidden-layer SCReNN (as defined in Eq. (68)) is a separator of m pattern sets $\mathcal{X}_i$ such that the i^{th} output y_i satisfies $y_i(\mathbf{x}) > 0$ for any $\mathbf{x} \in \mathcal{X}_i$ and $y_i(\mathbf{x}) < 0$ for any $\mathbf{x}$ in the other pattern sets. Hence

$$y_i(\mathbf{x}) = \mathbf{a}_i^T \max(0, W_2^T \max\{0, W_1^T\mathbf{x} + \mathbf{b}_1\} + \mathbf{b}_2) + \mathbf{c} \tag{73}$$

is a binary two-hidden-layer sign constrained ReNN separator for $\mathcal{X}_i$ and the set $\bigcup_{j\neq i} \mathcal{X}_j$. By Lemma 8, one can decompose the union $\bigcup_{j\neq i} \mathcal{X}_j$ into a number of subsets that are convexly-separable from $\mathcal{X}_i$. Then, for each such subset, namely $\hat{\mathcal{X}}$, of the union, one can use a single-hidden-layer SCReNN to separate $\hat{\mathcal{X}}$ and $\mathcal{X}_i$, and decomposed $\mathcal{X}_i$ into several subsets which are linearly separable from $\hat{\mathcal{X}}$. One can use the linear separators of these decomposed subsets to analyse the data and the key discriminant factors.

5. Training of Sign Constrained Rectifier Networks

In this section, we first introduce the well known MM algorithm for non-convex optimization problems, and then show how the convexity/concavity properties of SCRNs can be used to design convex surrogate functions in order to apply the MM algorithms to learn the parameters of the proposed SCRNs.

5.1. The MM Algorithm

The MM algorithm is an iterative algorithm for minimization of non-convex objective functions, with each of its iterations consisting of two steps: the majorization step which finds a surrogate function that upperbounds the objective function, and the minimization step which minimizes the surrogate function. Let $f(\mathbf{x})$ be a real-valued function, $\mathbf{x}^{(l)}$ represent a fixed value of the parameter $\mathbf{x}$, and let $g(\mathbf{x}; \mathbf{x}^{(l)})$ denote a real-valued function of $\mathbf{x}$ whose form depends on $\mathbf{x}^{(l)}$. The function $g(\mathbf{x}; \mathbf{x}^{(l)})$ is said to majorize a real-valued function $f(\mathbf{x})$ at the point $\mathbf{x}^{(l)}$ if

$$\left\{ \begin{array}{rcl} f(\mathbf{x}) & \leq & g(\mathbf{x}; \mathbf{x}^{(l)}), \ \forall\, \mathbf{x} \in \Omega \\ f(\mathbf{x}^{(l)}) & = & g(\mathbf{x}^{(l)}; \mathbf{x}^{(l)}) \end{array} \right. \tag{74}$$

where $\Omega \subset \mathbb{R}^n$ is a closed convex set. The general idea of the MM procedure to solve the minimization problem

$$\min_{\mathbf{x}\in\Omega} f(\mathbf{x}) \tag{75}$$

is to construct a majorization function $g(\mathbf{x}; \mathbf{x}^{(l)})$ and update $\mathbf{x}$ at iteration l by

$$\mathbf{x}^{(l+1)} = \arg\min_{\mathbf{x}\in\Omega} g(\mathbf{x}; \mathbf{x}^{(l)}). \tag{76}$$

In practice, convex majorization functions are usually used so that the minimization problem at each iteration has no local minima and can be solved efficiently. It is easy to show that the above iterative scheme decreases the objective function monotonically in each iteration, i.e.,

$$f(\mathbf{x}^{(l+1)}) \leq g(\mathbf{x}^{(l+1)}; \mathbf{x}^{(l)}) \leq g(\mathbf{x}^{(l)}; \mathbf{x}^{(l)}) = f(\mathbf{x}^{(l)}), \tag{77}$$

where the first inequality and the last equality follow from (74) while the sandwiched inequality follows from (76). Hence the objective function decreases until it converges to a stationary point. Moreover, many local minima of the objective function can be avoided if they have larger values than the minimum of one of the majorization functions in the iterations of the MM algorithm. Hence, the MM algorithm usually finds a good solution even though it cannot guarantee to find the global minima.

Next, we use the MM algorithm to address the training of single hidden layer SCRNs.

5.2. Training of Single Hidden Layer SCRNs

Consider the following single hidden layer SCRN

$$\begin{aligned} f(\mathbf{x};\mathcal{W}) &= b_0 - \mathbf{1}^T \max\{0, W^T\mathbf{x}+\mathbf{b}\} \\ &= b_0 - \textstyle\sum_{k=1}^m \max(0, \mathbf{w}_k^T\mathbf{x}+b_k) \end{aligned} \tag{78}$$

for binary classification, where $\mathcal{W} \triangleq \{b_0, \mathbf{w}_k, b_k, k \in [m]\}$ is the set of weights and biases. By Lemma 1, $f(\mathbf{x};\mathcal{W})$ is a concave function of $\mathcal{W}$ when $\mathbf{x}$ is fixed. Therefore, the hinge loss of the positive patterns

$$J_+(\mathcal{W}) \triangleq \sum_{\mathbf{x}\in\mathcal{X}_+} \max\{0, 1 - f(\mathbf{x};\mathcal{W})\} \tag{79}$$

is a convex function of $\mathcal{W}$. Although the hinge loss of the negative patterns, namely

$$J_-(\mathcal{W}) \triangleq \sum_{\mathbf{x}\in\mathcal{X}_-} \max\{0, 1 + f(\mathbf{x};\mathcal{W})\} \tag{80}$$

is not convex, it is bounded by the following convex function

$$\widehat{J}_-(\mathcal{W};\mathcal{W}_0) \triangleq \sum_{\mathbf{x}\in\mathcal{X}_-} \max\{0, 1 + \widehat{f}(\mathbf{x};\mathcal{W},\mathcal{W}_0)\} \tag{81}$$

where $\mathcal{W}_0 = \{b_{0,0}, \mathbf{w}_{k,0}, b_{k,0}, k \in [m]\}$ is a fixed set of parameters, and

$$\begin{aligned} \widehat{f}(\mathbf{x};\mathcal{W},\mathcal{W}_0) &\triangleq b_0 - \sum_{k\in\mathcal{K}(\mathbf{x},\mathcal{W}_0)} \mathbf{w}_k^T\mathbf{x} + b_k \\ &\geq b_0 - \sum_{k\in[m]} \max\{0, \mathbf{w}_k^T\mathbf{x}+b_k\} \\ &= f(\mathbf{x};\mathcal{W}) \\ \mathcal{K}(\mathbf{x},\mathcal{W}_0) &\triangleq \left\{k : 1 \leq k \leq m, \mathbf{w}_{k,0}^T\mathbf{x} + b_{k,0} > 0\right\}. \end{aligned} \tag{82}$$

Therefore the total hinge loss satisfies the following inequality

$$\begin{aligned} J(\mathcal{W}) &= J_+(\mathcal{W}) + J_-(\mathcal{W}) \\ &\leq J_+(\mathcal{W}) + \widehat{J}_-(\mathcal{W};\mathcal{W}_0). \end{aligned} \tag{83}$$

That is, $J(\mathcal{W})$ is bounded by a convex function of $\mathcal{W}$, i.e., $J_+(\mathcal{W}) + \widehat{J}_-(\mathcal{W};\mathcal{W}_0)$.

Hence, the minimization problem of the hinge loss $J(\mathcal{W})$, with some convex regularization term $R(\mathcal{W})$, can be solved with the efficient MM algorithm as below

$$\mathcal{W}^{(l+1)} = \arg\min_{\mathcal{W}} \quad R(\mathcal{W}) + J_+(\mathcal{W}) + \widehat{J}_-(\mathcal{W};\mathcal{W}^{(l)}). \tag{84}$$

At each iteration, one needs to solve a convex minimization problem and the cost function decreases until convergence, i.e.,

$$\begin{aligned} &R(\mathcal{W}^{(l+1)}) + J_+(\mathcal{W}^{(l+1)}) + J_-(\mathcal{W}^{(l+1)}) \\ &\quad\leq R(\mathcal{W}^{(l)}) + J_+(\mathcal{W}^{(l)}) + J_-(\mathcal{W}^{(l)}). \end{aligned} \tag{85}$$

In our experiments, we implement the MM algorithm and update the weights of the neural networks as follows. **First**, we compute the derivatives of the majorized cost function at the current state, namely

$$\Delta\mathcal{W}^{(l)} \triangleq \left.\frac{d\hat{J}(\mathcal{W};\mathcal{W}^{(l)})}{d\mathcal{W}}\right|_{\mathcal{W}=\mathcal{W}^{(l)}} \tag{86}$$

where $\hat{J}(\mathcal{W};\mathcal{W}^{(l)}) = R(\mathcal{W}) + J_+(\mathcal{W}) + \hat{J}_-(\mathcal{W};\mathcal{W}^{(l)})$. **We then** solve the following one-dimensional minimization problem

$$\begin{aligned} \alpha^* &= \arg\min_{\alpha} f_l(\alpha) \\ f_l(\alpha) &\triangleq \hat{J}(\mathcal{W}^{(l)} + \alpha\Delta\mathcal{W}^{(l)};\mathcal{W}^{(l)}) \end{aligned} \tag{87}$$

and update $\mathcal{W}$ as

$$\mathcal{W}^{(l+1)} = \mathcal{W}^{(l)} + \alpha^*\Delta\mathcal{W}^{(l)}. \tag{88}$$

Since $f_l(\alpha)$ is convex, it has a unique minima α^*. Furthermore, $f_l(\alpha)$ is monotonically decreasing when $\alpha \leq \alpha^*$ and monotonically increasing when $\alpha \geq \alpha^*$. Using this property, we apply the following bisection method to find the unique minima α^*: **1)** we first set $\alpha_* = 0$ and find a sufficiently large α_1 such that $f_l(\alpha_1 + \eta) > f_l(\alpha_1)$ where η is a small step size and we set $\eta = 0.0001$ in our implementation; **2)** let $\alpha = (\alpha_* + \alpha_1)/2$ and compute $f(\alpha)$ and $f(\alpha + \eta)$. Update $\alpha_1 = \alpha$ if $f(\alpha + \eta) > f(\alpha)$ and update $\alpha^* = \alpha$ otherwise. This procedure continues until $|f(\alpha_1) - f(\alpha^*)| \leq \epsilon$ where ϵ is a threshold which is set to 0.0001 in our implementation.

Compared to the conventional gradient-descent based optimization methods which requires small step sizes for convergence, the proposed MM algorithm can find the optimal step size to speed up the convergence at each iteration. Our experimental results are provided in Section 6 to show the efficiency of the proposed MM algorithm.

Next, we consider the training of SCRNs with two hidden layers.

5.3. Training of Two-Hidden-Layer SCRNs

Let

$$\begin{aligned} f(\mathbf{x};\mathcal{W}) &= b_0 - \mathbf{1}^T \max(0, \mathbf{z}_2) \\ \mathbf{z}_2 &= W_2^T \max(0, \mathbf{z}_1) + \mathbf{b}_2 \\ \mathbf{z}_1 &= W_1^T\mathbf{x} + \mathbf{b}_1 \\ W_2 &\preceq 0 \end{aligned} \tag{89}$$

be a two-hidden-layer SCRN. We learn its weight matrices and bias vectors by minimizing the following cost function

$$\begin{aligned} J(\mathcal{W}) &\triangleq R(\mathcal{W}) + J_+(\mathcal{W}) + J_-(\mathcal{W}) \\ J_+(\mathcal{W}) &\triangleq \sum_{\mathbf{x}\in\mathcal{X}_+} \max\{0, 1 - f(\mathbf{x};\mathcal{W})\} \\ J_-(\mathcal{W}) &\triangleq \sum_{\mathbf{x}\in\mathcal{X}_-} \max\{0, 1 + f(\mathbf{x};\mathcal{W})\} \end{aligned} \tag{90}$$

where $\mathcal{W} \triangleq \{b_0, W_1, W_2, \mathbf{b}_1, \mathbf{b}_2\}$ is the set of the parameters in the network, $R(\mathcal{W})$ is a convex regularisation term, $\mathcal{X}_+$ and $\mathcal{X}_-$ are the pattern sets with labels 1 and -1 respectively.

When the first layer weight matrix W_1 and the bias vector $\mathbf{b}_1$ are fixed, the learning of the other parameters in $\mathcal{W}$ is essentially a training problem of a single hidden layer SCRN and thus can be optimized by using the algorithm presented in Section 5.2. Next, we consider the optimization of $(W_1, \mathbf{b}_1)$ when the other parameters are fixed. The following Lemma provides the foundation for the algorithm to be presented.

Lemma 10. *Let $\mathbf{a}_1 \in \{0,1\}^{l_1}, \mathbf{a}_2 \in \{0,1\}^{l_2}$ be two arbitrary activation patterns for the first layer nodes and for the second layer nodes respectively. Denote*

$$\begin{aligned} f_1(\mathbf{x}; \mathcal{W}, \mathbf{a}_1) &= b_0 - \mathbf{1}^T \max(0, \hat{\mathbf{z}}_2) \\ \hat{\mathbf{z}}_2 &= W_2^T \mathrm{diag}\{\mathbf{a}_1\}\mathbf{z}_1 + \mathbf{b}_2 \\ \mathbf{z}_1 &= W_1^T \mathbf{x} + \mathbf{b}_1 \end{aligned} \tag{91}$$

and

$$\begin{aligned} f_2(\mathbf{x}; \mathcal{W}, \mathbf{a}_2) &= b_0 - \mathbf{a}_2^T \mathbf{z}_2 \\ \mathbf{z}_2 &= W_2^T \max(0, \mathbf{z}_1) + \mathbf{b}_2 \\ \mathbf{z}_1 &= W_1^T \mathbf{x} + \mathbf{b}_1. \end{aligned} \tag{92}$$

Then we have

(i) $f_1(\mathbf{x}; \mathcal{W}, \mathbf{a}_1)$ is a concave function of $(W_1, \mathbf{b}_1)$ when the other parameters in $\mathcal{W}$ are fixed, and furthermore

$$f_1(\mathbf{x}; \mathcal{W}, \mathbf{a}_1) \le f(\mathbf{x}; \mathcal{W}). \tag{93}$$

(ii) $f_2(\mathbf{x}; \mathcal{W}, \mathbf{a}_2)$ is a convex function of $(W_1, \mathbf{b}_1)$ when the other parameters in $\mathcal{W}$ are fixed, and furthermore

$$f_2(\mathbf{x}; \mathcal{W}, \mathbf{a}_2) \ge f(\mathbf{x}; \mathcal{W}). \tag{94}$$

Proof: Note that $\hat{\mathbf{z}}_2$ is a linear function of $(W_1, \mathbf{b}_1)$ and $-\mathbf{1} \preceq 0$. From Lemma 1, it follows that $f_1(\mathbf{x}; \mathcal{W}, \mathbf{a}_1)$ is a concave function of $(W_1, \mathbf{b}_1)$. Furthermore, since $W_2 \preceq 0$ and $\mathrm{diag}\{\mathbf{a}_1\}\mathbf{z}_1 \le \max(0, \mathbf{z}_1)$, we have $\hat{\mathbf{z}}_2 \succeq \mathbf{z}_2$ and therefore Eq. (93) holds. This proves the first statement (i).

For the proof of statement (ii), Eq. (94) is true due to the fact that $\mathbf{a}_2^T \mathbf{z}_2 \le \mathbf{1}^T \max(0, \mathbf{z}_2)$. To prove the convexity of $f_2(\mathbf{x}; \mathcal{W}, \mathbf{a}_2)$ as a function of $(W_1, \mathbf{b}_1)$, let $\hat{b}_0 = b_0 - \mathbf{a}_2^T \mathbf{b}_2$ and $\hat{\mathbf{a}}_2^T = -\mathbf{a}_2^T W_2 \succeq 0$. Then $f_2(\mathbf{x}; \mathcal{W}, \mathbf{a}_2) = \hat{b}_0 + \hat{\mathbf{a}}_2 \max(0, W_1^T \mathbf{x} + \mathbf{b}_1)$. Therefore, by Lemma 1, $f_2(\mathbf{x}; \mathcal{W}, \mathbf{a}_2)$ is a convex function of $(W_1, \mathbf{b}_1)$ when $\mathbf{x}$ and the other parameters in $\mathcal{W}$ are fixed. This proves the statement (ii) and completes the proof.

□

Based on Lemma 10, $(W_1, \mathbf{b}_1)$ can be optimized iteratively using the MM algorithm as follows. Let $\mathcal{W}^{(l)}$ be the parameter set at step l. $\mathcal{W}^{(0)}$ can be any arbitrary initialization. Let $\mathbf{a}_{1,l}(\mathbf{x})$ and $\mathbf{a}_{2,l}(\mathbf{x})$ be the activation patterns of $\mathbf{z}_1(\mathbf{x})$ and $\mathbf{z}_2(\mathbf{x})$ respectively at step l, and denote

$$\begin{aligned} \widehat{J}_+(\mathcal{W}, \mathcal{W}^{(l)}) &\triangleq \sum_{\mathbf{x} \in \mathcal{X}_+} \max\{0, 1 - f_1(\mathbf{x}; \mathcal{W}, \mathbf{a}_{1,l}(\mathbf{x}))\} \\ \widehat{J}_-(\mathcal{W}, , \mathcal{W}^{(l)}) &\triangleq \sum_{\mathbf{x} \in \mathcal{X}_-} \max\{0, 1 + f_2(\mathbf{x}; \mathcal{W}, \mathbf{a}_{2,l}(\mathbf{x}))\}. \end{aligned} \tag{95}$$

Then, from Lemma 10, we have

$$\begin{aligned} J_+(\mathcal{W}) &\le \widehat{J}_+(\mathcal{W}, \mathcal{W}^{(l)}) \\ J_-(\mathcal{W}) &\le \widehat{J}_-(\mathcal{W}, \mathcal{W}^{(l)}). \end{aligned} \tag{96}$$

Furthermore, $\widehat{J}_+(\mathcal{W}, \mathcal{W}^{(l)})$ and $\widehat{J}_-(\mathcal{W}, \mathcal{W}^{(l)})$ are convex functions of $(W_1, \mathbf{b}_1)$ when the other parameters are fixed. Hence $\mathcal{W}$ can be updated as

$$\mathcal{W}^{(l+1)} = \arg \min_{W_1, \mathbf{b}_1} \quad R(\mathcal{W}) + \widehat{J_+}(\mathcal{W}; \mathcal{W}^{(l)}) + \widehat{J_-}(\mathcal{W}; \mathcal{W}^{(l)}). \tag{97}$$

From Eq. (96), the cost function is strictly decreasing until convergence, that is

$$\begin{aligned} &R(\mathcal{W}^{(l+1)}) + J_+(\mathcal{W}^{(l+1)}) + J_-(\mathcal{W}^{(l+1)}) \\ &\quad \le R(\mathcal{W}^{(l)}) + J_+(\mathcal{W}^{(l)}) + J_-(\mathcal{W}^{(l)}). \end{aligned} \tag{98}$$

The whole set of parameters $\mathcal{W}$ can be learnt by optimizing $(b_0, W_2, \mathbf{b}_2)$ and $(W_1, \mathbf{b}_1)$ alternatively.

6. Experimental Results

Experiments were conducted on the MNIST database of handwritten digits [45] to demonstrate the benefit of sign constraints and the effectiveness of the proposed algorithm. The MNIST dataset consists of 28×28 grey scale images of handwritten digits. The total number of images in the dataset is 70,000, of which 50,000 are used for training, 10,000 for validation and the remaining 10,000 are used for testing. These images belong to 10 different classes (corresponding to digits from 0 to 9).

In our experiments, we used two baseline deep learning networks, referred by baseline (BL) and batch normalization (BN) respectively, from MatConvNet [46] for MNIST database. BL is a eight-layer neural network consisting of two convolution neural networks each followed by a max-pooling layer, a fully connected layer followed by a rectifier layer and an output layer followed by a softmax layer. BN includes all the layers of BL but three batch normalization layers are added after the rectifier layer and the two max-pooling layers. To show the benefits of sign-constraints, we restricted the output layer to have non-positive elements and restricted the fully connected layer to have non-negative constraints. The performances of the sign-constrained neural networks, namely BL+SC and BN+SC, are reported in Table 1 with comparisons to the performances of the baseline neural networks BL and BN. The baseline networks and the sign-constrained networks are all trained for 50 epochs using the same learning rates. The best model is selected by using the validation performances and the corresponding training and testing performances are reported.

	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**%

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.

Table 1 shows that the sign constraints consistently improve classification accuracy for both validation and testing and for both BL and BN.

Next, we show the benefit of the sign constraints in the understanding of the geometrical properties of the separating boundaries of the learnt neural network BN+SC. Since the output layer has non-positive weights, the learnt classifier is a sign constrained neural network with a single hidden layer if we treat the features of the second last hidden layer as inputs. Since it has 500 nodes, the learnt classifier is the minimum of up to 2^{500} linear classifiers in the feature space. We applied the efficient subset decomposition algorithm presented in Section 3.3.1 and approximated the learnt classifier with the minimum of up to 50 linear classifiers. The performances of the approximations with different numbers of linear classifiers are reported in Figure 1. With 10 linear classifiers, the test error rate is below 1% and with 40 linear classifiers, the test error rate is 0.76%, which is close 0.75%, the performance of the original classifier. This shows that the separating boundaries in the feature space can be well approximated by 40 linear classifiers.

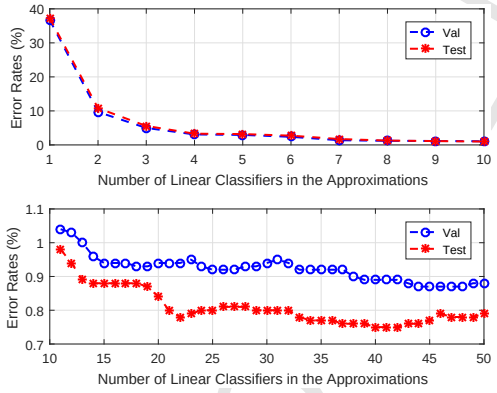


Figure 1: Performances Versus Number of Linear Classifiers used in the Approximations of the Sign Constrained Neural Network.

To show the advantages of the proposed MM algorithms, we conducted experiments on a binary classification problem, which separates digit 0 from the other nine digits on the MNIST database. At each iteration, we used the convexity of the majorized cost function to find the best step size using the bisection method proposed in Section 5.2. Figure 3 shows the optimal step sizes of the proposed MM algorithm in the training, and Figure 2 shows its fast convergence. With the optimal step sizes, the MM algorithm achieves error rates below 1% within 3 iterations for all the training, validation and testing performances. After 10 iterations, the training error rate drops to 0.1%, while the validation and test performances drop to 0.21% and 0.31% respectively.

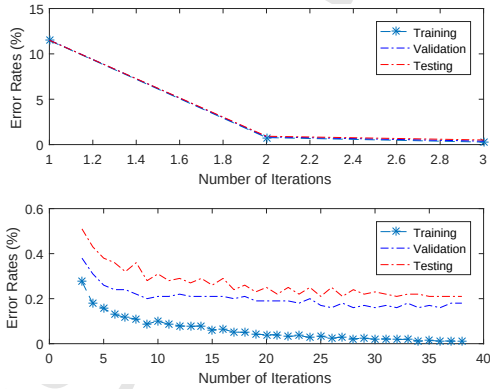


Figure 2: Convergence of MM algorithms on the MNIST Database. The error rates of first three iterations are shown in the top figure.

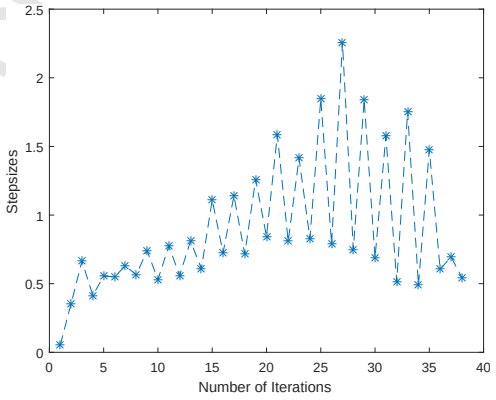


Figure 3: Adaptive Step Sizes of the MM algorithm on the MNIST Database.

7. Concluding Remarks

We have shown that, with sign constraints on the weights of the output and the second hidden layers, two-hidden-layer SCRNs are still universal classifiers and are capable of decomposing each class of patterns into several subsets so that each subset is convexly separable from the other pattern set. In addition, single-hidden-layer SCRNs are capable of separating any two (or more) convexly separable pattern sets as well as decomposing one of them into several subsets so that each subset is linearly separable from the other pattern set. The proposed SCRN not only enables pattern and feature analysis for model interpretability and knowledge discovery but also enables efficient training with the well known MM algorithms to reduce the risks of local minima. Experimental results demonstrate that the sign constraints also improve classification accuracy. Future potential research directions include investigating the applications of this work to dynamic neural networks for feedback control systems.

Acknowledgements

The authors would like to thank the reviewers for their valuable comments and helpful suggestions. This work was supported by the Australian Research Council under Grant DP150100294, Grant DP150104251, and Grant DE120102960.

References

[1] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, 2012, pp. 1097–1105.

[2] M. D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: Computer Vision–ECCV 2014, Springer, 2014, pp. 818–833.

[3] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, Z. Tu, Deeply-supervised nets, in: Artificial Intelligence and Statistics, 2015, pp. 562–570.

[4] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1026–1034.

[5] G. Huang, Z. Liu, K. Q. Weinberger, L. van der Maaten, Densely connected convolutional networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, Vol. 1, 2017, p. 3.

[6] Y. Taigman, M. Yang, M. Ranzato, L. Wolf, Deepface: Closing the gap to human-level performance in face verification, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 1701–1708.

[7] Y. Sun, Y. Chen, X. Wang, X. Tang, Deep learning face representation by joint identification-verification, in: Advances in Neural Information Processing Systems, 2014, pp. 1988–1996.

[8] F. Seide, G. Li, D. Yu, Conversational speech transcription using context-dependent deep neural networks., in: Interspeech, 2011, pp. 437–440.

[9] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al., Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, IEEE Signal Processing Magazine 29 (6) (2012) 82–97.

[10] L. Deng, J. Li, J.-T. Huang, K. Yao, D. Yu, F. Seide, M. Seltzer, G. Zweig, X. He, J. Williams, et al., Recent advances in deep learning for speech research at microsoft, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2013, pp. 8604–8608.

[11] D. Ciresan, U. Meier, J. Schmidhuber, Multi-column deep neural networks for image classification, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2012, pp. 3642–3649.

[12] T. Poston, C.-N. Lee, Y. Choie, Y. Kwon, Local minima and back propagation, in: International Joint Conference on Neural Networks (IJCNN), Vol. 2, IEEE, 1991, pp. 173–176.

[13] R. Livni, S. Shalev-Shwartz, O. Shamir, On the computational efficiency of training neural networks, in: Advances in Neural Information Processing Systems, 2014, pp. 855–863.

[14] B. D. Haeffele, R. Vidal, Global optimality in tensor factorization, deep learning, and beyond, arXiv preprint arXiv:1506.07540, 2015.

[15] D. Soudry, Y. Carmon, No bad local minima: Data independent training error guarantees for multilayer neural networks, arXiv preprint arXiv:1605.08361, 2016.

[16] M. Soltanolkotabi, A. Javanmard, J. D. Lee, Theoretical insights into the optimization landscape of over-parameterized shallow neural networks, arXiv preprint arXiv:1707.04926, 2017.

[17] Q. Nguyen, M. Hein, The loss surface of deep and wide neural networks, arXiv preprint arXiv:1704.08045, 2017.

[18] D. Boob, G. Lan, Theoretical properties of the global optimizer of two layer neural network, arXiv preprint arXiv:1710.11241, 2017.

[19] K. Kawaguchi, Deep learning without poor local minima, in: Advances in Neural Information Processing Systems, 2016, pp. 586–594.

[20] I. Safran, O. Shamir, Spurious local minima are common in two-layer relu neural networks, arXiv preprint arXiv:1712.08968, 2017.

[21] P. Auer, M. Herbster, M. K. Warmuth, Exponentially many local minima for single neurons, in: Advances in neural information processing systems, 1996, pp. 316–322.

[22] R. Ge, F. Huang, C. Jin, Y. Yuan, Escaping from saddle pointsonline stochastic gradient for tensor decomposition, in: Conference on Learning Theory, 2015, pp. 797–842.

[23] R. Ge, J. D. Lee, T. Ma, Matrix completion has no spurious local minimum, in: Advances in Neural Information Processing Systems, 2016.

[24] J. Sun, Q. Qu, J. Wright, When are nonconvex problems not scary?, arXiv preprint arXiv:1510.06096, 2015.

[25] S. Bhojanapalli, B. Neyshabur, N. Srebro, Global optimality of local search for low rank matrix recovery, in: Advances in Neural Information Processing Systems, 2016, pp. 3873–3881.

[26] B. Ristera, D. L. Rubin, Piecewise convexity of artificial neural networks, Neural Networks 94 (2017) 34–45.

[27] Y. Sun, P. Babu, D. P. Palomar, Majorization-minimization algorithms in signal processing, communications, and machine learning, IEEE Transactions on Signal Processing 65 (3) (2017) 794–816.

[28] C. Wathes, H. H. Kristensen, J.-M. Aerts, D. Berckmans, Is precision livestock farming an engineer's daydream or nightmare, an animal's friend or foe, and a farmer's panacea or pitfall?, Computers and Electronics in Agriculture 64 (1) (2008) 2–10.

[29] P. J. Hepworth, A. V. Nefedov, I. B. Muchnik, K. L. Morgan, Broiler chickens can benefit from machine learning: support vector machine analysis of observational epidemiological data, Journal of The Royal Society Interface 9 (73) (2012) 1934–1942.

[30] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural networks 2 (5) (1989) 359–366.

[31] N. Le Roux, Y. Bengio, Deep belief networks are compact universal approximators, Neural computation 22 (8) (2010) 2192–2207.

[32] G. Montufar, N. Ay, Refinements of universal approximation results for deep belief networks and restricted Boltzmann machines, Neural Computation 23 (5) (2011) 1306–1319.

[33] S. An, F. Boussaid, M. Bennamoun, How can deep rectifier networks achieve linear separability and preserve distances?, in: Proceedings of the 32nd International Conference on Machine Learning (ICML-15), 2015, pp. 514–523.

[34] S. An, Q. Ke, M. Bennamoun, F. Boussaid, F. Sohel, Sign constrained rectifier networks with applications to pattern decompositions, in: Machine Learning and Knowledge Discovery in Databases, Springer, 2015, pp. 546–559.

[35] M. Gupta, L. Jin, N. Homma, Static and dynamic neural networks: from fundamentals to advanced theory, John Wiley & Sons, 2004.

[36] Y. Wei, J. H. Park, H. R. Karimi, Y.-C. Tian, H. Jung, Improved stability and stabilization results for stochastic synchronization of continuous-time semi-markovian jump neural networks with time-varying delay, IEEE transactions on neural networks and learning systems, 2017.

[37] Y. Wei, J. Qiu, H. R. Karimi, Reliable output feedback control of discrete-time fuzzy affine systems with actuator faults, IEEE Transactions on Circuits and Systems I: Regular Papers 64 (1) (2017) 170–181.

[38] J. Qiu, H. R. Karimi, et al., Fuzzy-affine-model-based memory filter design of nonlinear systems with time-varying delay, IEEE Transactions on Fuzzy Systems, 2017.

[39] Y. Wei, J. H. Park, J. Qiu, H. Jung, Reliable output feedback control for piecewise affine systems with markov-type sensor failure, IEEE Transactions on Circuits and Systems II: Express Briefs, 2017.

[40] Y. Xu, R. Lu, P. Shi, J. Tao, S. Xie, Robust estimation for neural networks with randomly occurring distributed delays and markovian jump coupling, IEEE transactions on neural networks and learning systems, 2017.

[41] Y. Xu, Z. Wang, D. Yao, R. Lu, C.-Y. Su, State estimation for periodic neural networks with uncertain weight matrices and markovian jump channel states, IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2017.

[42] Y. Wang, H. Shen, D. Duan, On stabilization of quantized sampled-data neural-network-based control systems, IEEE transactions on cybernetics 47 (10) (2017) 3124–3135.

[43] Y. Wang, H. Shen, H. R. Karimi, D. Duan, Dissipativity-based fuzzy integral sliding mode control of continuous-time TS fuzzy systems, IEEE Transactions on Fuzzy Systems, 2017.

[44] Y. Wang, Y. Xia, H. Shen, P. Zhou, SMC design for robust stabilization of nonlinear markovian jump singular systems, IEEE Transactions on Automatic Control 63 (1) (2018) 219–224.

[45] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE, 1998.

[46] A. Vedaldi, K. Lenc, Matconvnet – convolutional neural networks for matlab, in: Proceeding of the ACM Int. Conf. on Multimedia, 2015.