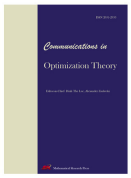
Commun. Optim. Theory 2027 (2027) 17 https://doi.org/10.23952/cot.2027.17

MATHRES Communications in Optimization Theory
Available online at http://cot.mathres.org



# KKT-BASED OPTIMALITY CONDITIONS FOR NEURAL NETWORK APPROXIMATION

VINESHA PEIRIS[1], NADEZDA SUKHORUKOVA[2,*], JULIEN UGON[3]

[1]Centre for Optimisation and Decision Science, Curtin University, Australia
[2]Department of Mathematics, Swinburne University of Technology, Australia
[3]School of Information Technology, Deakin University, Australia

**Abstract.** In this paper, we obtain necessary optimality conditions for neural network approximation. We consider neural networks in Manhattan ($l_1$ norm) and Chebyshev (max norm). The activation functions are restricted to the class of smooth functions. The optimality conditions are based on neural networks with at most one hidden layer. We reformulate nonsmooth unconstrained optimisation problems as larger dimension constrained problems with smooth objective functions and constraints. Then we use KKT conditions to develop the necessary conditions and present the optimality conditions in terms of convex analysis and convex sets.

**Keywords.** KKT condition; Min-max approximation; Neural networks; Optimality conditions.

**2020 Mathematics Subject Classification.** 41A50, 26A27, 65D10.

## 1. INTRODUCTION

The theory of Artificial Neural Networks (ANNs) or just neural networks appeared as an extension of work by Kolmogorov and Arnold [2, 8], who were working on the 13th problem of Hilbert. Their celebrated representation theorem and further developments lead to an efficient adaptation to function approximation, established in [3, 7, 9, 11]. A detailed overview of optimisation approaches for studying neural networks can be found in [12].

In this paper, we approach neural networks purely from the point of view of optimisation. Therefore, we are minimising the amount of "purely" neural network vocabulary. Essentially, neural networks provide accurate approximations to (continuous) functions. In most practical problems, function values are defined at a finite number of points (discretisation points).

Our approach is to formulate the corresponding approximation problems as optimisation problems. These problems are nonsmooth and nonconvex. We reformulate these nonsmooth problems as equivalent smooth ones by introducing extra variables and constraints (standard

*Corresponding author.
E-mail address: m.peiris@curtin.edu.au (V. Peiris), nsukhorukova@swin.edu.au (N. Sukhorukova), julien.ugon@deakin.edu.au (J. Ugon).

Received December 12, 2025; Accepted March 31, 2026.

©2027 Communications in Optimization Theory

approach) and then formulate the corresponding KKT conditions, that are necessary optimality conditions, but, in general, they are not sufficient.

The rest of the paper is organised as follows. In Section 2, we present the necessary knowledge from the theory of neural networks and formulate the corresponding optimisation problems. Then in Section 3, we formulate the corresponding KKT conditions and interpret them from the point of view of convex analysis. Finally, in Section 4, we present the conclusions and possible further research directions.

## 2. PRELIMINARIES

2.1. **Neural networks and approximation.** Neural networks can be seen as approximation prototypes. In this paper, we focus on the simplest neural networks (with at most one hidden layer). Without too many details, the structure of the corresponding approximations is as follows:

(1) single hidden layer (one hidden layer):

$$\sum_{j=1}^{n} a_j \sigma((w_1^j, w_2^j, \dots, w_d^j)\mathbf{T_i} + w_0^j), \tag{2.1}$$

(2) no hidden layer (zero hidden layers, index $j$ is redundant):

$$\sigma((w_1, w_2, \dots, w_d)\mathbf{T_i} + w_0). \tag{2.2}$$

In these formulations, $\mathbf{T_i} \in \mathbb{R}^{\mathbf{d}}$, $i = 1, \dots, N$ are the discretisation points and

$$a_1, \dots, a_n, w_0^j, w_1^j \dots, w_d^j, \ j = 1, \dots, n$$

are the approximation parameters ($w_0^j, w_1^j \dots, w_d^j$, $j = 1, \dots, n$ are also known as network weights), $\sigma : \mathbb{R} \to \mathbb{R}$ is the activation function (univariate).

**Remark 2.1.** In the case of no hidden layers (2.2), we assume for simplicity that $a_1 = 1$. This coefficient is "absorbed" by the activation function without changing the smoothness. Since in (2.2) there is only one set of weights, we omit the upper index for simplicity. The case when $a = -1$ can be reduced to the approximation of the function $-f(x)$.

The approximation parameters are the decision variable of the corresponding optimisation problems. It is clear that the number of decision variables is significantly higher in the case of one hidden layer.

The activation function is predefined and not subject to optimisation. A common choice for activation functions are sigmoidal functions, ReLU and Leaky ReLU [6]. In [7] and [11] the authors showed that any nonpolynomial activation function can be used for accurate approximation constructions.

The goal is to minimise a loss function, which is simply an error function. The approximation errors (deviations) are calculated at the discretisation points and then combined in the chosen loss function. Common choices of loss functions are:

(1) least squares (sum of squares of deviation at discretisation points, $L_2^2$);
(2) manhattan loss function (sum of absolute deviations of errors, $L_1$);
(3) uniform loss function (maximum of absolute deviations of errors, $L_{\max}$).

Least squares is a popular choice: in the case of smooth activation functions, the corresponding optimisation problems are smooth. In this paper, however, we focus on nonsmooth loss functions, namely, $L_1$ and $L_{\max}$. We also assume that the activation function is smooth, leaving ReLU and Leaky ReLU for future research directions. In the case of nonsmooth activation functions, one has to apply nonsmooth optimisation techniques [4, 5], since the gradients of the functions appearing in the KKT conditions are not defined due to their nonsmoothness.

**Remark 2.2.** In [10] the authors obtained equivalent necessary optimality conditions for the uniform-based loss function (no hidden layer) using nonsmooth optimisation techniques. In the current paper, we show that these conditions can be obtained using the KKT conditions (without applying nonsmooth optimisation techniques) when the activation function is smooth. In addition, in this paper, we develop the optimality conditions for $L_1$.

2.2. **Uniform-based loss function.** The optimisation problem is as follows:

$$\text{minimise} \max_{\substack{\mathbf{T}_i \\ i=1,\dots,N}} \left| \sum_{j=1}^{n} a_j \sigma(\mathbf{w}^j \mathbf{T}_i + w_0^j) - f(\mathbf{T}_i) \right|, \tag{2.3}$$

$$\text{subject to } X \in \mathbb{R}^{n+nd+n}, \tag{2.4}$$

where

- $N$ is the number of discretisation points,
- $\mathbf{T}_i \in Q$, $i = 1,\dots,N$ are $d$-dimensional vectors (dataset points) where $Q$ is a $d$-dimensional hypercube,
- $n$ is the maximal number of components in the sum (that is, the number of nodes in the hidden layer), $n = 1$ in the case of the absence of the hidden layers (index $j$ is redundant) and $n > 1$ for the case with one hidden layer,
- vectors $\mathbf{w}^j = (w_1^j,\dots,w_d^j)$ are the weights where $w_k^j$, $j = 1,\dots,n$, $k = 0,\dots,d$, and coefficients $a_j$, $j = 1,\dots,n$ are the decision variables. All the decision variables are placed in a single vector,

$$X = (a_1,\dots,a_n,w_1^1,\dots,w_1^n,\dots,w_d^1,\dots,w_d^n,w_0^1,\dots,w_0^n),$$

- $\sigma$ is the activation function and,
- $f$ is the original function (function to approximate).

An equivalent formulation of (2.3)-(2.4), where all the objective function and the constraint functions are smooth, is as follows:

$$\text{minimise } z \tag{2.5}$$

subject to

$$\sum_{j=1}^{n} a_j \sigma(\mathbf{w}^j \mathbf{T}_i + w_0^j) - f(\mathbf{T}_i) \le z \tag{2.6}$$

$$f(\mathbf{T}_i) - \sum_{j=1}^{n} a_j \sigma(\mathbf{w}^j \mathbf{T}_i + w_0^j) \le z \tag{2.7}$$

$$X \in \mathbb{R}^{n+nd+n},\ i = 1,\dots,N. \tag{2.8}$$

The objective function and the constraint functions are smooth and therefore KKT conditions can be used to formulate the necessary optimality conditions. One more decision variable $z$ was introduced. The total number of constraints is $2N$, which is large in most practical problems.

**Remark 2.3.** In the case of no hidden layers and monotonous activation function the problem is quasiconvex and can be efficiently solved using, for example, a bisection method [13].

2.3. **Manhattan-based loss function.** The optimisation problem is as follows:

$$\text{minimise } \sum_{i=1}^{N} \left| \sum_{j=1}^{n} a_j \sigma(\mathbf{w}^j \mathbf{T}_i + w_0^j) - f(\mathbf{T}_i) \right|, \tag{2.9}$$

$$\text{subject to } X \in \mathbb{R}^{n+nd+n}, \tag{2.10}$$

where the components are the same as in section 2.2.

An equivalent formulation of (2.9)-(2.10), where all the objective function and the constraint functions are smooth, is as follows:

$$\text{minimise } \sum_{i=1}^{N} z_i \tag{2.11}$$

subject to

$$\sum_{j=1}^{n} a_j \sigma(\mathbf{w}^j \mathbf{T}_i + w_0^j) - f(\mathbf{T}_i) \leq z_i \tag{2.12}$$

$$f(\mathbf{T}_i) - \sum_{j=1}^{n} a_j \sigma(\mathbf{w}^j \mathbf{T}_i + w_0^j) \leq z_i \tag{2.13}$$

$$X \in \mathbb{R}^{n+nd+n},\ i = 1, \ldots, N. \tag{2.14}$$

Similar to the uniform-based loss function case, the objective function and the constraint functions are smooth and therefore KKT conditions can be used to formulate necessary optimality conditions. Extra $N$ decision variable $z_i$, $i = 1, \ldots, N$ were introduced. The total number of constraints is $2N$.

In the next section, we consider the simplest cases of neural networks: no hidden layer and one single layer. At first glance, this is an obvious limitation. However, it is important to start with simpler cases. Moreover, due to [3], even one hidden layer lead to accurate approximations. The only requirement is to choose nonpolynomial functions as activation functions [9, 11]. At the same time, the number of hidden nodes may be large [1].

## 3. Neural Networks Approximation

In this section, we re-numerate the indices of the discretisation points. This is done in order to express the optimality conditions in terms of positive and negative deviation. As we will see shortly, in the case of uniform-based approximation, the optimality conditions are based on maximal absolute deviation points only, while in the case of Manhattan-based approximation, one has to take into account all the discretisation points, including the point where the deviation is zero.

The renumeration is as follows:

- indices $i = 1,\dots,n_1$, $i \in \mathbb{P}$ correspond to positive maximal absolute deviation points, such that
$$\sigma(\mathbf{w}^j\mathbf{T}_i + w_0^j) - f(\mathbf{T}_i) > 0;$$
- indices $i = n_1+1,\dots,n_1+n_2$, $i \in \mathbb{N}$ correspond to negative maximal absolute deviation points, such that
$$\sigma(\mathbf{w}^j\mathbf{T}_i + w_0^j) - f(\mathbf{T}_i) < 0;$$
- remaining indices $i = n_1+n_2+1,\dots,n_1+n_2+n_3$, $i \in \mathbb{C}$ correspond to the points whose deviation is zero.

Note that $n_1+n_2+n_3 = N$.

### 3.1. **Uniform-based loss function.**

3.1.1. *No hidden layer.* In this case, the decision variables are
$$(w_0, w_1, \dots, w_d, z).$$
Then KKT conditions are as follows:
$$\begin{bmatrix}0\\0\\\vdots\\0\\1\end{bmatrix} + \sum_{i=1}^{n_1}\lambda_i\begin{bmatrix}\begin{bmatrix}1\\\mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w}\mathbf{T}_i + w_0)\\-1\end{bmatrix} + \sum_{i=n_1+1}^{n_1+n_2}\lambda_i\begin{bmatrix}\begin{bmatrix}-1\\-\mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w}\mathbf{T}_i + w_0)\\-1\end{bmatrix} = \mathbf{0}_{d+2}, \tag{3.1}$$
where
$$\lambda_i \geq 0,\ i = 1,\dots,n_1+n_2. \tag{3.2}$$
and $\sigma'$ is the derivative of the univariate activation function $\sigma$. Note that due to the complementarity conditions,
$$\lambda_i = 0,\ i = n_1+n_2+1,\dots,n_1+n_2+n_3.$$
From the last row of (3.1) one can see that
$$\sum_{i=1}^{n_1+n_2}\lambda_i = 1,$$
while from the first row
$$\sum_{i=1}^{n_1}\lambda_i = \sum_{i=n_1+1}^{n_1+n_2}\lambda_i = 0.5.$$
Therefore, KKT can be written as follows:
$$\operatorname{co}\left\{\begin{bmatrix}1\\\mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w}\mathbf{T}_i + w_0),\ i \in \mathbb{P}\right\} \cap \operatorname{co}\left\{\begin{bmatrix}1\\\mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w}\mathbf{T}_i + w_0),\ i \in \mathbb{N}\right\} \neq \emptyset. \tag{3.3}$$
The condition (3.3) is a necessary optimality condition. Therefore, in the case of neural network approximation with the following characteristics:

- uniform loss function (Chebyshev approximation);
- smooth activation function;
- no hidden layer,

the following theorem holds.

**Theorem 3.1.** *If a neural network approximation is optimal, then the convex polytopes*

$$\operatorname{co}\left\{\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w}\mathbf{T}_i+w_0),\ i\in\mathbb{P}\right\}\ \textit{and}\ \operatorname{co}\left\{\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w}\mathbf{T}_i+w_0),\ i\in\mathbb{N}\right\} \tag{3.4}$$

*have a common point, where $\mathbb{P}$ and $\mathbb{N}$ are the sets of the maximal deviation points (positive and negative, respectively).*

3.1.2. *One hidden layer.* In this case, the decision variables are

$$(w_0^1, w_1^1, \dots, w_d^1, \dots w_0^n, w_1^n, \dots, w_d^n, a_1, \dots, a_n, z),$$

where $a_1, \dots, a_n$ are the coefficients from the representation (2.1). Then KKT conditions are as follows:

$$\begin{bmatrix}0\\0\\ \vdots\\0\\1\end{bmatrix} + \sum_{i=1}^{n_1}\lambda_i \begin{bmatrix} a_1\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_1}\mathbf{T}_i+w_0)\\ \vdots \\ a_n\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_n}\mathbf{T}_i+w_0)\\ \begin{bmatrix}\sigma(\mathbf{w}_1\mathbf{T}_i+w_1^0)\\ \vdots\\ \sigma(\mathbf{w}_n\mathbf{T}_i+w_n^0)\end{bmatrix}\\ -1\end{bmatrix} + \sum_{i=n_1+1}^{n_1+n_2}\lambda_i \begin{bmatrix} a_1\begin{bmatrix}-1\\ -\mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_1}\mathbf{T}_i+w_0)\\ \vdots \\ a_n\begin{bmatrix}-1\\ -\mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_n}\mathbf{T}_i+w_0)\\ \begin{bmatrix}-\sigma(\mathbf{w}_1\mathbf{T}_i+w_1^0)\\ \vdots\\ -\sigma(\mathbf{w}_n\mathbf{T}_i+w_n^0)\end{bmatrix}\\ -1\end{bmatrix} = \mathbf{0}_{d+2+n}, \tag{3.5}$$

$$\lambda_i \geq 0,\ i=1,\dots,n_1+n_2. \tag{3.6}$$

Similarly to the situation without a hidden layer, using the last coordinate, one can see that

$$\sum_{i=1}^{n_1+n_2}\lambda_i = 1.$$

Therefore, a necessary optimality condition can be presented as follows:

$$\operatorname{co}\left\{\begin{bmatrix} a_1\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_1}\mathbf{T}_i+w_0)\\ \vdots \\ a_n\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_n}\mathbf{T}_i+w_0)\\ \begin{bmatrix}\sigma(\mathbf{w}_1\mathbf{T}_i+w_1^0)\\ \vdots\\ \sigma(\mathbf{w}_n\mathbf{T}_i+w_n^0)\end{bmatrix}\end{bmatrix},\ i\in\mathbb{P}\right\} \cap \operatorname{co}\left\{\begin{bmatrix} a_1\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_1}\mathbf{T}_i+w_0)\\ \vdots \\ a_n\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_n}\mathbf{T}_i+w_0)\\ \begin{bmatrix}\sigma(\mathbf{w}_1\mathbf{T}_i+w_1^0)\\ \vdots\\ \sigma(\mathbf{w}_n\mathbf{T}_i+w_n^0)\end{bmatrix}\end{bmatrix},\ i\in\mathbb{N}\right\} \neq \emptyset, \tag{3.7}$$

where set $\mathbb{P}$ is the set of positive maximal absolute deviation points, while set $\mathbb{N}$ is the set of negative maximal absolute deviation points.

Therefore, in the case of neural network approximation with the following characteristics:

- uniform loss function (Chebyshev approximation);
- smooth activation function;
- one hidden layer,

the following theorem holds.

**Theorem 3.2.** *If a neural network approximation is optimal, then the convex polytopes*

$$\operatorname{co}\left\{\begin{bmatrix} a_1\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_1}\mathbf{T}_i+w_0)\\ \vdots \\ a_n\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_n}\mathbf{T}_i+w_0)\\ \begin{bmatrix}\sigma(\mathbf{w}_1\mathbf{T}_i+w_1^0)\\ \vdots\\ \sigma(\mathbf{w}_n\mathbf{T}_i+w_n^0)\end{bmatrix}\end{bmatrix},\ i\in\mathbb{P}\right\}\ \textit{and}\ \operatorname{co}\left\{\begin{bmatrix} a_1\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_1}\mathbf{T}_i+w_0)\\ \vdots \\ a_n\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w_n}\mathbf{T}_i+w_0)\\ \begin{bmatrix}\sigma(\mathbf{w}_1\mathbf{T}_i+w_1^0)\\ \vdots\\ \sigma(\mathbf{w}_n\mathbf{T}_i+w_n^0)\end{bmatrix}\end{bmatrix},\ i\in\mathbb{N}\right\} \tag{3.8}$$

*have a common point, where $\mathbb{P}$ and $\mathbb{N}$ are the sets of the maximal deviation points (positive and negative, respectively).*

Similarly to the case without hidden layers, the necessary optimality conditions are based on maximal absolute deviation points. The situation in the case of one hidden layer is significantly more complex.

3.2. **Manhattan-based loss function.** For Manhattan-based approximation, we use the following notation:

- $\mathbf{E}_k$ is a $k$-th dimensional column vector, whose components are equal to 1;
- $\mathbf{O}_k$ is a $k$-th dimensional column vector, whose components are equal to 0;
- $\mathbf{J}_{ki}$ is a $k$-th dimensional column vector, whose components are equal to 0, while the $i$-th component is equal to 1.

3.2.1. *No hidden layer.* In this case, the decision variables are

$$(w_0, w_1, \ldots, w_d, z_1, \ldots, z_N).$$

Then KKT conditions are as follows:

$$\begin{bmatrix}0\\0\\ \vdots\\0\\ \mathbf{E}_{n_1}\\ \mathbf{E}_{n_2}\\ \mathbf{E}_{n_3}\end{bmatrix}+\sum_{i=1}^{n_1}\lambda_i\begin{bmatrix}\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w}\mathbf{T}_i+w_0)\\ -\mathbf{J}_{n_1 i}\\ \mathbf{O}_{n_2}\\ \mathbf{O}_{n_3}\end{bmatrix}+\sum_{i=n_1+1}^{n_1+n_2}\lambda_i\begin{bmatrix}\begin{bmatrix}-1\\ -\mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w}\mathbf{T}_i+w_0)\\ \mathbf{O}_{n_1}\\ -\mathbf{J}_{n_2 i}\\ \mathbf{O}_{n_3}\end{bmatrix}+\mathbf{D}=\mathbf{0}_{d+1+N}, \tag{3.9}$$

where

$$\mathbf{D}=\sum_{i=n_1+n_2+1}^{N}\lambda_i\begin{bmatrix}\begin{bmatrix}1\\ \mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w}\mathbf{T}_i+w_0)\\ \mathbf{O}_{n_1}\\ \mathbf{O}_{n_2}\\ \mathbf{J}_{n_3 i}\end{bmatrix}+\sum_{i=1+n_1+n_2+n_3}^{n_1+n_2+2n_3}\lambda_i\begin{bmatrix}\begin{bmatrix}-1\\ -\mathbf{T}_i\end{bmatrix}\sigma'(\mathbf{w}\mathbf{T}_i+w_0)\\ \mathbf{O}_{n_1}\\ \mathbf{O}_{n_2}\\ \mathbf{J}_{n_3 i}\end{bmatrix}, \tag{3.10}$$

$\sigma'$ is the derivative of the univariate activation function $\sigma$,

$$\lambda_i\geq 0,\ i=1,\ldots,n_1+n_2+2n_3. \tag{3.11}$$

Therefore,

$$\lambda_i = 1,\ i = 1,\dots,n_1+n_2$$

and

$$\lambda_{n_1+n_2+i} + \lambda_{n_1+n_2+n_3+i} = 1,\ i = 1,\dots,n_3.$$

Then in the case of neural network approximation with the following characteristics:

- Manhattan loss function;
- smooth activation function;
- no hidden layer,

the following theorem holds.

**Theorem 3.3.** *If an approximation is optimal, then*

$$\sum_{i=1}^{n_1} \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}\mathbf{T}_i + w_0) - \sum_{i=n_1+1}^{n_1+n_2} \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}\mathbf{T}_i + w_0) \in \mathbb{Q}, \tag{3.12}$$

*where*

$$\mathbb{Q} = \sum_{i=n_1+n_2+1}^{N} \operatorname{co}\left\{ \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}\mathbf{T}_i + w_0), - \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}\mathbf{T}_i + w_0) \right\}. \tag{3.13}$$

In the case of no discretisation point, where the deviation is zero, set $\mathbb{Q} = \mathbf{0}_{d+1}$ and the necessary condition can be simplified. In this case, the polytopes that characterise the KKT conditions are singletons and hence they are much simpler to verify.

**Corollary 3.4.** *In the case of no zero-deviation points, the necessary optimality conditions are*

$$\sum_{i=1}^{n_1} \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}\mathbf{T}_i + w_0) = \sum_{i=n_1+1}^{n_1+n_2} \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}\mathbf{T}_i + w_0).$$

3.2.2. *One hidden layer.* In this case, the decision variables are

$$(w_0^1, w_1^1, \dots, w_d^1, \dots w_0^n, w_1^n, \dots, w_d^n, a_1 \dots, a_n, z_1, \dots, z_N).$$

Then KKT conditions are as follows:

$$\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \mathbf{E}_{n_1} \\ \mathbf{E}_{n_2} \\ \mathbf{E}_{n_3} \end{bmatrix} + \sum_{i=1}^{n_1} \lambda_i \begin{bmatrix} a_1 \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^1\mathbf{T}_i + w_0^1) \\ \vdots \\ a_n \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^n\mathbf{T}_i + w_0^n) \\ \begin{bmatrix} \sigma(\mathbf{w}^1\mathbf{T}_i + w_0^1) \\ \vdots \\ \sigma(\mathbf{w}^n\mathbf{T}_i + w_0^n) \end{bmatrix} \\ -\mathbf{J}_{n_1 i} \\ \mathbf{O}_{n_2} \\ \mathbf{O}_{n_3} \end{bmatrix} + \sum_{i=n_1+1}^{n_1+n_2} \lambda_i \begin{bmatrix} a_1 \begin{bmatrix} -1 \\ -\mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^1\mathbf{T}_i + w_0^1) \\ \vdots \\ a_n \begin{bmatrix} -1 \\ -\mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^n\mathbf{T}_i + w_0^n) \\ \begin{bmatrix} -\sigma(\mathbf{w}_1^T\mathbf{T}_i + w_1^0) \\ \vdots \\ -\sigma(\mathbf{w}_n^T\mathbf{T}_i + w_n^0) \end{bmatrix} \\ \mathbf{O}_{n_1} \\ -\mathbf{J}_{n_2 i} \\ \mathbf{O}_{n_3} \end{bmatrix} + \mathbf{D} = \mathbf{0}_{d+1+N+n}, \tag{3.14}$$

where

$$\mathbf{D} = \sum_{i=n_1+n_2+1}^{N} \lambda_i \begin{bmatrix} a_1 \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^1\mathbf{T}_i + w_0^1) \\ \vdots \\ a_n \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^n\mathbf{T}_i + w_0^n) \\ \begin{bmatrix} \sigma(\mathbf{w}^1\mathbf{T}_i + w_1^0) \\ \vdots \\ \sigma(\mathbf{w}^n\mathbf{T}_i + w_n^0) \end{bmatrix} \\ \mathbf{O}_{n_1} \\ \mathbf{O}_{n_2} \\ \mathbf{J}_{n_3 i} \end{bmatrix} + \sum_{i=1+n_1+n_2+n_3}^{n_1+n_2+2n_3} \lambda_i \begin{bmatrix} a_1 \begin{bmatrix} -1 \\ -\mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^1\mathbf{T}_i + w_0^1) \\ \vdots \\ a_n \begin{bmatrix} -1 \\ -\mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^n\mathbf{T}_i + w_0^n) \\ \begin{bmatrix} -\sigma(\mathbf{w}^1\mathbf{T}_i + w_0^1) \\ \vdots \\ -\sigma(\mathbf{w}^n\mathbf{T}_i + w_0^n) \end{bmatrix} \\ \mathbf{O}_{n_1} \\ \mathbf{O}_{n_2} \\ \mathbf{J}_{n_3 i} \end{bmatrix}, \tag{3.15}$$

$$\lambda_i \geq 0,\ i = 1, \ldots, n_1 + n_2 + 2n_3. \tag{3.16}$$

Therefore,

$$\lambda_i = 1,\ i = 1, \ldots, n_1 + n_2$$

and

$$\lambda_{n_1+n_2+i} + \lambda_{n_1+n_2+n_3+i} = 1,\ i = 1, \ldots, n_3.$$

Then in the case of neural network approximation with the following characteristics:

- Manhattan loss function;
- smooth activation function;
- one hidden layer,

the following theorem holds.

**Theorem 3.5.** *If an approximation is optimal, then*

$$\sum_{i=1}^{n_1} \begin{bmatrix} a_1 \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^1\mathbf{T}_i + w_0^1) \\ \vdots \\ a_n \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^n\mathbf{T}_i + w_0^n) \\ \begin{bmatrix} \sigma(\mathbf{w}^1\mathbf{T}_i + w_0^1) \\ \vdots \\ \sigma(\mathbf{w}^n\mathbf{T}_i + w_0^n) \end{bmatrix} \end{bmatrix} + \sum_{i=n_1+1}^{n_1+n_2} \begin{bmatrix} a_1 \begin{bmatrix} -1 \\ -\mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^1\mathbf{T}_i + w_0^1) \\ \vdots \\ a_n \begin{bmatrix} -1 \\ -\mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^n\mathbf{T}_i + w_0^n) \\ \begin{bmatrix} -\sigma(\mathbf{w}^1\mathbf{T}_i + w_0^1) \\ \vdots \\ -\sigma(\mathbf{w}^n\mathbf{T}_i + w_0^n) \end{bmatrix} \end{bmatrix} \in \mathbb{Q}, \tag{3.17}$$

*where*

$$\mathbb{Q} = \sum_{i=n_1+n_2+1}^{N} \begin{bmatrix} \operatorname{co}\left\{ \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} a_1 \sigma'(\mathbf{w}^1 \mathbf{T}_i + w_0), - \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} a_1 \sigma'(\mathbf{w}^1 \mathbf{T}_i + w_0) \right\} \\ \vdots \\ \operatorname{co}\left\{ \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} a_n \sigma'(\mathbf{w}^n \mathbf{T}_i + w_0), - \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} a_n \sigma'(\mathbf{w}^n \mathbf{T}_i + w_0) \right\} \\ \sigma(\mathbf{w}^1 \mathbf{T}_i + w_0^1) \operatorname{co}\{1,-1\} \\ \vdots \\ \sigma(\mathbf{w}^1 \mathbf{T}_i + w_0^1) \operatorname{co}\{1,-1\} \end{bmatrix}. \tag{3.18}$$

Similarly to the case without hidden layers, the conditions can be simplified if none of the discretisation points corresponds to zero deviation.

**Corollary 3.6.** *In the case of no zero-deviation points, the necessary optimality conditions are*

$$\sum_{i=1}^{n_1} \begin{bmatrix} a_1 \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^1 \mathbf{T}_i + w_0^1) \\ \vdots \\ a_n \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^n \mathbf{T}_i + w_0^n) \\ \begin{bmatrix} \sigma(\mathbf{w}^1 \mathbf{T}_i + w_0^1) \\ \vdots \\ \sigma(\mathbf{w}^n \mathbf{T}_i + w_0^n) \end{bmatrix} \end{bmatrix} = \sum_{i=n_1+1}^{n_1+n_2} \begin{bmatrix} a_1 \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^1 \mathbf{T}_i + w_0^1) \\ \vdots \\ a_n \begin{bmatrix} 1 \\ \mathbf{T}_i \end{bmatrix} \sigma'(\mathbf{w}^n \mathbf{T}_i + w_0^n) \\ \begin{bmatrix} \sigma(\mathbf{w}^1 \mathbf{T}_i + w_0^1) \\ \vdots \\ \sigma(\mathbf{w}^n \mathbf{T}_i + w_0^n) \end{bmatrix} \end{bmatrix}.$$

Therefore, similar to the case without hidden layers (no zero deviation discretisation points), the polytopes that characterise the KKT conditions are singletons.

## 4. Conclusions

In this paper, we developed necessary optimality conditions for neural network approximation for at most one hidden layer (uniform and Manhattan loss functions). The nonsmooth optimisation unconstrained problems are formulated as smooth optimisation problems with smooth constraints and therefore the corresponding KKT can be used.

In the case of the uniform loss function, the conditions are based on the overlapping of polytopes whose vertices correspond to the maximal deviation points. Essentially, only the maximal deviation points contribute to these conditions.

In the case of the Manhattan loss function, the situation is different. First of all, the optimality conditions, formulated in terms of convex polytopes, include all the discretisation points (not just maximal deviation points). Second, the conditions can be simplified in the case when none of the discretisation points corresponds to the deviation equal to zero: the corresponding polytopes are singletons.

One possible future research direction is to extend the study to nonsmooth activation functions (for example, ReLU and Leaky ReLU). Some research in this direction is done in [10].

## REFERENCES


[1] J.M. Almira, P.E. Lopez-de-Teruel, D.J. Romero-López, F. Voigtlaender, Negative results for approximation using single layer and multilayer feedforward neural networks, J. Math. Anal. Appl. 494 (2021) 124584.

[2] V. Arnold, On functions of three variables. Dokl. Akad. Nauk SSSR 114 (1957) 679–681. (English translation: Amer. Math. Soc. Transl. 28 (1963) 51–54.)

[3] G. Cybenko, Approximation by superpositions of a sigmoidal function, Math. Control, Signals Sys. 2 (1989) 303–314.

[4] V. F. Demyanov and L. C. W. Dixon, Quasidifferential calculus, Mathematical programming study, Elsevier, North-Holland, 1986.

[5] V. F. Demyanov and V. N. Malozemov, Introduction to Minimax, Moscow, 1971.

[6] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016. `http://www.deeplearningbook.org`.

[7] K. Hornik, Approximation capabilities of multilayer feedforward networks, Neural Networks, 4 (1991) 251–257.

[8] A. N. Kolmogorov, On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition, Dokl. Akad. Nauk SSSR, 114 (1957) 953–956.

[9] M. Leshno, Vladimir Ya. Lin, A Pinkus, and S Schocken, Multilayer feedforward networks with a nonpolynomial activation function can approximate any function, Neural Networks, 6 (1993) 861–867.

[10] V. Peiris, N. Sukhorukova, Nonsmooth Optimisation and neural networks, arXiv:2503.01860v1 [math.OC] https://doi.org/10.48550/arXiv.2503.01860

[11] A. Pinkus, Approximation theory of the MLP model in neural networks, Acta Numerica, 8 (1999) 143–195.

[12] R. Sun, Optimization for deep learning: An overview, J. Oper. Res. Soc. Chin. 8 (2020) 249-294.

[13] V. Peiris, V. Roshchina and N. Sukhorukova, Artificial neural networks with uniform norm-based loss functions, Adv. Comput. Math. 50 (2024) 31.