Neural Networks 131 (2020) 29–36

Contents lists available at ScienceDirect

Neural Networks

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

Two-hidden-layer feed-forward networks are universal approximators: A constructive approach

Eduardo Paluzo-Hidalgo [a,*], Rocio Gonzalez-Diaz [a], Miguel A. Gutiérrez-Naranjo [b]

[a] Department of Applied Mathematics I, University of Seville, Seville, Spain
[b] Department of Computer Science and Artificial Intelligence, University of Seville, Seville, Spain

ARTICLE INFO

Article history:
Received 12 August 2019
Received in revised form 11 May 2020
Accepted 14 July 2020
Available online 22 July 2020

Keywords:
Universal Approximation Theorem
Simplicial Approximation Theorem
Multi-layer feed-forward network
Triangulations

ABSTRACT

It is well-known that artificial neural networks are universal approximators. The classical existence result proves that, given a continuous function on a compact set embedded in an n-dimensional space, there exists a one-hidden-layer feed-forward network that approximates the function. In this paper, a constructive approach to this problem is given for the case of a continuous function on triangulated spaces. Once a triangulation of the space is given, a two-hidden-layer feed-forward network with a concrete set of weights is computed. The level of the approximation depends on the refinement of the triangulation.

1. Introduction

One of the first results in the development of neural networks is the *Universal Approximation Theorem* (Cybenko, 1989; Hornik, 1991). This classical result shows that any continuous function on a compact set in $\mathbb{R}^n$ can be approximated by a multi-layer feed-forward network with only one hidden layer and non-polynomial activation function (like the sigmoid function). It is well-known that this result has two important drawbacks for its practical use: firstly, the width of the hidden layer grows exponentially and, secondly, the proofs developed in Cybenko (1989) and Hornik (1991) do not provide a practical algorithm for building such a network. Bearing these results in mind, many researchers are paying attention to theoretical aspects of the current success of neural network architectures and searching for bounds for the depth and width of such networks and the possibility that they act as universal approximators (see, e.g., Liang & Srikant, 2016; Safran & Shamir, 2017; Telgarsky, 2016 among many others).

Undoubtedly, the use of many hidden layers is a big contribution to the success of deep learning architectures (Sun, Chen, Wang, Liu, & Liu, 2016), but instead of exploring the power of depth, recently several studies have made interesting contributions about the power of width (Hanin & Sellke, 2017; Lu, Pu, Wang, Hu, & Wang, 2017; Nguyen, Mukkamala, & Hein, 2018). To sum up, these works show that there exist continuous functions on compact sets that cannot be approximated by any neural network if the width of the layers is not larger than a bound, regardless of the depth of the network. In Guliyev and Ismailov (2018), the authors constructively proved that single-hidden-layer feed-forward networks with fixed weights are universal approximators for univariate functions, and they provided a step-by-step construction. However, as the authors claimed, not all continuous multivariate functions can be approximated by such neural networks. Their case of study could be considered somehow a particular case of our approach for the 1-dimensional case.

Other interesting research line on the expressive power of neural networks follows an algebraic approach. In Delalleau and Bengio (2011), Martens and Medabalimi (2014) and Poon and Domingos (2012), sum product networks are explored and tensor properties are studied in Cohen, Sharir, and Shashua (2016) and Cohen and Shashua (2016). A different perspective was introduced in Kileel, Trager, and Bruna (2019) where "deep polynomial neural networks" (where the activation function is a polynomial exponentiation) are considered.

In this paper, we provide an effective method for finding the weights of a two-hidden-layer feed-forward network which approximates a given continuous function between two triangulable metric spaces. Let us remark that the method is constructive, and it only depends on the desired level of approximation to the given function. Our approach is based on a classical result from algebraic topology. Roughly speaking, our result is based on two observations: Firstly, triangulable spaces can be "modeled" using simplicial complexes, and a continuous function between

* Corresponding author.
E-mail addresses: epaluzo@us.es (E. Paluzo-Hidalgo), rogodi@us.es (R. Gonzalez-Diaz), magutier@us.es (M.A. Gutiérrez-Naranjo).
URLs: https://personal.us.es/epaluzo (E. Paluzo-Hidalgo), https://personal.us.es/rogodi (R. Gonzalez-Diaz), http://www.cs.us.es/~naranjo/ (M.A. Gutiérrez-Naranjo).

https://doi.org/10.1016/j.neunet.2020.07.021
0893-6080/

two triangulable spaces can be approximated by a simplicial map between simplicial complexes. Secondly, a simplicial map between simplicial complexes can be "modeled" as a two-hidden-layer feed-forward network. Let us remark that the classical result Universal Approximation Theorem is valid for all compact sets on $\mathbb{R}^n$ and our results presented here are valid for triangulable spaces. However, triangulable spaces are common in real-world problems. Furthermore, we would like to highlight that an advantage of our approach from a theoretical and practical point of view is that it can be useful to solve real-world problems that can be modeled by triangulable spaces.

The paper is organized as follows: In Section 2, the preliminary notions about multi-layer feed-forward networks and simplicial complexes are provided. Then, in Section 3, a concrete architecture of such networks that acts equivalently to a given simplicial map is given. In Sections 4 and 5, we extend the Simplicial Approximation Theorem and Universal Approximation Theorem, respectively. The complexity of the architecture is studied in Section 6. A specific example is described in Section 7. Finally, conclusions are given in Section 8.

2. Background

In this section, we recall some notions about artificial neural networks and simplicial complexes.

2.1. Multi-layer feed-forward networks

Artificial neural networks are inspired by biological networks of alive neurons in a brain. The number of different architectures, algorithms, and areas of application have recently grown in many directions. In general, a neural network can be formalized as a function $\mathcal{N}_{\omega,\Theta} : \mathbb{R}^n \to \mathbb{R}^m$ that depends on a set of weights ω and a set of parameters Θ which involves the description of activation functions, layers, synapses between nodes (neurons), and whatever other consideration in its architecture. A good introduction to artificial neural networks can be found in Haykin (1999).

In this paper, we focus on one of the simplest classes of artificial neural networks: multi-layer feedforward networks. They consist of three or more fully connected layers of nodes: an input layer, an output layer, and one or more hidden layers. Each node in one layer has an activation function and it is connected with every node in the following layer. The next definition formalizes this idea.

Definition 1 (*Adapted From Hornik, 1991*)**.** A multi-layer feed-forward network defined on a real-valued n-dimensional space is a function $\mathcal{N} : \mathbb{R}^n \to \mathbb{R}^m$ such that, for each $x \in \mathbb{R}^n$, $\mathcal{N}(x)$ is the composition of $k+1$ functions

$$\mathcal{N}(x) = f_{k+1} \circ f_k \circ \ldots \circ f_1(x)$$

where $k \in \mathbb{Z}$ is the number of hidden layers, $k \geq 1$, and, for $1 \leq i \leq k+1$, $f_i : \mathbb{R}^{d_{i-1}} \to \mathbb{R}^{d_i}$ is defined as

$$f_i(y) = \phi_i(W^{(i)}; y; b_i)$$

being $W^{(i)}$ a real-valued $d_{i-1} \times d_i$ matrix, (that is, $W^{(i)} \in \mathcal{M}_{d_{i-1}\times d_i}$), $b_i \in \mathbb{R}^{d_i}$ the bias term, and ϕ_i a bounded, continuous, and non-constant function (called activation function). Notice that $d_0 = n$, $d_{k+1} = m$ and $d_i \in \mathbb{Z}$, $1 \leq i \leq k$, is called the width of the ith hidden layer.

Next, we rewrite one of the most important theoretical results of multi-layer feed-forward networks adapted to our notation.

Theorem 1 (*Universal Approximation Theorem, Hornik, 1991*)**.** *Let A be any compact subset of $\mathbb{R}^n$ and let $C(A)$ be the space of real-valued continuous functions on A. Then, given any $\epsilon > 0$ and any function $g \in C(A)$, there exists a multi-layer feed-forward network $\mathcal{N} : \mathbb{R}^n \to \mathbb{R}$ approximating g, that is, $\|g - \mathcal{N}\| < \epsilon$.*

As far as we know, the existing proofs of this theorem are non-constructive. See Cybenko (1989), Hornik (1991) and Hornik, Stinchcombe, and White (1989) where it is claimed that there exists a one-hidden-layer feed-forward network $\mathcal{N} : \mathbb{R}^n \to \mathbb{R}$, defined as $\mathcal{N}(x) = f_2 \circ f_1(x)$ with $f_1(y) = \phi_1(W^{(1)}; y; b_1)$ and $f_2(y) = W^{(2)}y$, approximating g, that is, $\|g - \mathcal{N}\| < \epsilon$, but no general algorithm to build $\mathcal{N}$ is given. In this paper, we will provide a constructive approach to Theorem 1 through a two-hidden-layer feed-forward network. The most important restriction to our approach is that our result is valid for triangulable spaces instead of compact ones but, as pointed out above, triangulable spaces cover most of the real-world problems.

2.2. Simplicial complexes

In this subsection, we recall the main result used in this paper based on a classical theorem from algebraic topology known as the *Simplicial Approximation Theorem*. For a further comprehension of the field, Ayala, Domínguez, and Quintero (2002), Boissonnat, Chazal, and Yvinec (2018), Edelsbrunner and Harer (2010), Hatcher (2002) and Munkres (1984) can be consulted.

Simplicial complexes are a data structure widely used to represent topological spaces. They are versatile mathematical objects that decompose a given topological space into pieces called simplices.

Definition 2. Let $v_0, v_1, \ldots, v_i$ be $i+1$ affinely independent points in $\mathbb{R}^n$ (being $i \leq n$). An i-simplex, $\sigma = (v_0, v_1, \ldots, v_i)$, is the convex set

$$\Big\{x \in \mathbb{R}^n \;\Big|\; x = \sum_{j=0}^{i} \lambda_j v_j \text{ with } \lambda_j \geq 0 \text{ and } \sum_{j=0}^{i} \lambda_j = 1\Big\}.$$

The points $v_0, v_1, \ldots, v_i$ are the vertices of σ. The dimension of σ is i. We say that σ' is a face of σ (denoted as $\sigma' \preceq \sigma$) if σ' is an i'-simplex, with $i' \leq i$, and whose vertices are also vertices of σ.

When several simplices are joint together, a more complex structure called simplicial complex is built. The following definition exhibits the way simplices can be glued together to obtain such a simplicial complex.

Definition 3. A simplicial complex K is a collection of simplices such that:

1. $\sigma \in K$ and $\sigma' \preceq \sigma$ implies $\sigma' \in K$;
2. $\sigma, \mu \in K$ implies $\sigma \cap \mu$ is either empty or a face of both.

If such collection is finite, then K is a finite simplicial complex (see Fig. 1).

The underlying space of K, denoted by $|K|$, is the union of the simplices of K together with the topology inherited from the ambient Euclidean space where the simplices are placed.

Definition 4. A simplex $\sigma \in K$ is maximal if it is not a face of any other simplex in K. The dimension of K, denoted by $\dim(K)$ is the largest of the dimensions of its maximal simplices. The simplicial complex K is pure if all the maximal simplices have dimension equal to $\dim(K)$.

A subcomplex L of K is a simplicial complex such that $L \subseteq K$. The skeleton of K is a particular subcomplex of K. Let us remark that the 0-skeleton of K is its vertex set.

Definition 5. The subcomplex of K consisting of all the simplices of K of dimension j or less is called the j-skeleton of K and it is denoted by $K^{(j)}$.

Next, we recall the concept of the star of a simplex σ in a simplicial complex K. Intuitively, it is the subcomplex of K whose maximal simplices share σ as a face.

Definition 6. Let K be a simplicial complex and σ a simplex of K. The star of σ in K, denoted by $\mathrm{st}(\sigma)$, is defined as:

$$\mathrm{st}(\sigma) = \left\{\mu \in K \;\middle|\; \exists\, \xi \in K \text{ such that } \sigma \preceq \xi \text{ and } \mu \preceq \xi\right\}.$$

As said above, simplicial complexes are combinatorial data structures used to model topological spaces. A way to obtain a refined model from an existing one is to subdivide it into small pieces so that the result is topologically equivalent to the former.

Definition 7. Let K and K' be simplicial complexes. It is said that K' is a subdivision of K if:

1. $|K| = |K'|$;
2. $\sigma' \in K'$ implies that there exists $\sigma \in K$ such that σ' is contained in σ.

The barycentric subdivision is a concrete example of subdivision of simplicial complexes. Using that the barycenter of an i-simplex $\sigma = (v_0 \dots v_i)$ is

$$b(\sigma) = \sum_{j=0}^{i} \frac{1}{j+1} v_j,$$

the definition of barycentric subdivision of a simplicial complex arises in a natural way.

Definition 8. Let K be a simplicial complex. The barycentric subdivision of the 0-skeleton of K is defined as the set of vertices of K, that is, $\mathrm{Sd}\, K^{(0)} = K^{(0)}$. Assuming we have $\mathrm{Sd}\, K^{(i-1)}$, which denotes the barycentric subdivision of the $(i-1)$-skeleton of K, $\mathrm{Sd}\, K^{(i)}$ is built by adding the barycenter of every i-simplex as a new vertex and connecting it to the simplices that subdivide the boundary of such i-simplex. The barycentric subdivision of K, denoted by $\mathrm{Sd}\, K$, is $\mathrm{Sd}\, K^{(u)}$ where u is the dimension of K. The iterated application of barycentric subdivisions is denoted by $\mathrm{Sd}^t K$ where t is the number of iterations (see Fig. 2).

Let us see now how the "geometric size" of the simplices of a simplicial complex can be measured.

Definition 9. Let K be a finite simplicial complex. The diameter of a simplex σ in K is defined as

$$\delta(\sigma) = \max\left\{\|x-y\| \;\text{ such that }\; x, y \text{ are vertices of } \sigma\right\}$$

and the mesh of K is defined as

$$m(K) = \max\left\{\delta(\sigma) \;\text{ such that }\; \sigma \in K\right\}.$$

Theorem 2 (Munkres, 1984, p. 86)**.** *Given a simplicial complex K and a real number $\epsilon > 0$, there exists an integer $t > 0$ such that $m(\mathrm{Sd}^t K) \leq \epsilon$.*

Let us now think about maps between simplicial complexes. These maps can be considered as extensions of simpler maps defined between the corresponding vertices of two given simplicial complexes. Interestingly, such maps can be considered as approximations of continuous functions defined on the underlying topological spaces that the simplicial complexes are modeling. Let us formalize these notions.

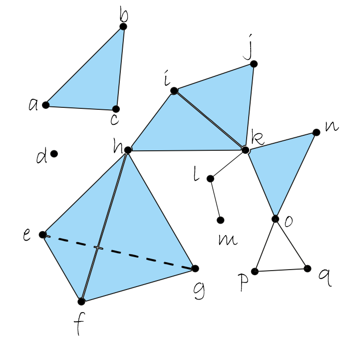


Fig. 1. Example of a simplicial complex. (d) is a 0-simplex, (o, p) is a 1-simplex, (a, b, c) is a 2-simplex, and (e, f, g, h) is a 3-simplex. The 1-simplex (a, b) is a face of (a, b, c). The maximal simplices of the simplicial complex are (e, f, g, h), (a, b, c), (i, h, k), (i, j, k), (o, p), (o, q), (p, q), and (d). The dimension of the simplicial complex is 3.

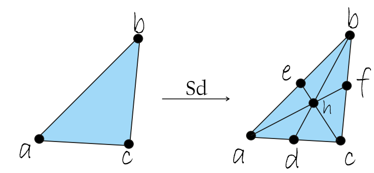


Fig. 2. On the left, a simplicial complex with just one maximal simplex, the 2-simplex (a, b, c). On the right, its first barycentric subdivision.

Definition 10. Let K and L be two simplicial complexes. A vertex map is a map $\varphi : K^{(0)} \to L^{(0)}$ with the property that for every simplex σ in K there exists a simplex μ in L such that the vertices of σ map to vertices of μ.

A vertex map φ can be extended to a continuous function $\varphi_c : |K| \to |L|$ in the following way.

Definition 11. Let K and L be two simplicial complexes and let $\varphi : K^{(0)} \to L^{(0)}$ be a vertex map. The simplicial map φ_c induced by φ is defined as follows. Let $x \in |K|$. Then there exist an i-simplex $\sigma = (v_0, \dots, v_i)$ in K and numbers $\lambda_j \geq 0$ such that

$$\sum_{j=0}^{i} \lambda_j = 1 \;\text{ and }\; x = \sum_{j=0}^{i} \lambda_j v_j.$$

Then

$$\varphi_c(x) = \sum_{j=0}^{i} \lambda_j \varphi(v_j).$$

Any vertex map φ induces a simplicial map φ_c, but if we want that map to be a simplicial approximation of a continuous function between the underlying spaces of two simplicial complexes K and L, a restriction on the star of each vertex of K must be added.

Definition 12. Let K and L be simplicial complexes and $g : |K| \to |L|$ a continuous function. A simplicial map $\varphi_c : |K| \to |L|$ induced by a vertex map $\varphi : K^{(0)} \to L^{(0)}$ is a simplicial approximation of g if

$$g(|\mathrm{st}(v)\,|) \subset |\mathrm{st}(\varphi(v))\,|$$

for each vertex v of K.

The Simplicial Approximation Theorem ensures the existence of simplicial maps that approximate continuous functions as close as we want.

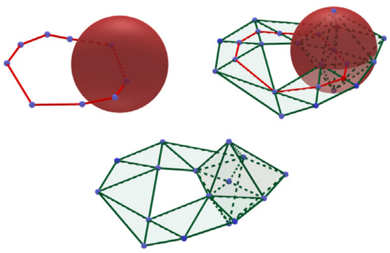


Fig. 3. From left to right and from top to bottom: (1) A sphere and a loop that intersect; (2) the sphere and the loop with a possible triangulation (the loop is covered by 2-simplices); (3) the triangulation of the loop and the sphere.

Theorem 3 (*Simplicial Approximation Theorem* Edelsbrunner & Harer, 2010, *p. 56*)**.** *If $g : |K| \to |L|$ is continuous then there is a sufficiently large integer $t > 0$ such that $\varphi_c : |\mathrm{Sd}^t K| \to |L|$ is a simplicial approximation of g.*

Theorems 1 and 3 are key results in their respective fields. Our aim in this paper is to consider Theorem 3 as a pillar to obtain a constructive approach to Theorem 1. Roughly speaking, the idea is to consider a simplicial approximation of a continuous function between two simplicial complexes. Such a simplicial approach is characterized through a vertex map which can be expressed as a neural network. Besides, the simplicial approximation can be chosen in such a way that it approximates the continuous function between the underlying spaces of two simplicial complexes. Since the parameters of the neural network can be effectively obtained from the vertex map, this method provides a constructive way to find a neural network that approximates a continuous function between the underlying spaces of two simplicial complexes. Following that aim, let us formalize the relationship between topological spaces and simplicial complexes.

Definition 13. A triangulation of a topological space X consists of a simplicial complex K and a homeomorphism $\tau : X \to |K|$. We say that the triangulation is finite if K is finite. We say that X is (finitely) triangulable if such (finite) triangulation exists.

The spaces that can be triangulated by simplicial complexes (see Hatcher, 2002, Theorem A.7, p. 525) are compact, locally contractible spaces that can be embedded in $\mathbb{R}^n$ for some n. Let us remark that the Universal Approximation Theorem (Theorem 1) is valid for any compact subset of $\mathbb{R}^n$, regardless of whether they are locally contractible or not. Not all compact sets in a metric space are locally contractible (see Geoghegan, 2010, Chapter 17.7, p. 426). Nevertheless, as far as we know, non-locally contractible spaces are very odd in $\mathbb{R}^n$ and this technical topological property has no practical application in real-world problems. Therefore, we can say that the results proved on triangulable spaces are true on a large amount of current neural network problems. Besides, we will restrict ourselves to finite pure triangulations. The other cases could be obtained using homotopies. For example, given a sphere and a circumference that intersect transversely, the sphere can be triangulated using 2-simplices, and the circumference can be covered by 2-simplices obtaining a finite pure triangulation of a space homotopic to the initial one (see Fig. 3).

Next, we extend the definition of a mesh of a simplicial complex in the following way.

Definition 14. Let X be a triangulable metric space and (K, τ) a finite triangulation of X. The mesh of X induced by (K, τ) is defined as

$$\tilde{m}_{(K,\tau)}(X) = \max\left\{\tilde{\delta}(\sigma) \mid \sigma \in K\right\}$$

where

$$\tilde{\delta}(\sigma) = \max\left\{d_X(x, y) \mid x = \tau^{-1}(a),\ \ y = \tau^{-1}(b);\, a, b \in \sigma\right\}$$

is the extended diameter of a simplex.

3. Multi-layer feed-forward networks and simplicial maps

In this section, we will show that simplicial maps can be modeled via multi-layer feed-forward networks in a straightforward way.

In the following theorem, we will compute a two-hidden-layer feed-forward network to model a simplicial map $\varphi_c : |K| \to |L|$ where K and L are finite pure simplicial complexes. This is not an important constraint in our case, since our final aim is to design a multi-layer feed-forward network that approximates a continuous function between finitely triangulable spaces.

Theorem 4. *Let us consider a simplicial map $\varphi_c : |K| \to |L|$ between the underlying space of two finite pure simplicial complexes K and L. Then a two-hidden-layer feed-forward network $\mathcal{N}_\varphi$ such that $\varphi_c(x) = \mathcal{N}_\varphi(x)$ for all $x \in |K|$ can be explicitly defined.*

Proof. Let us assume that $\dim(K) = n$ and $\dim(L) = m$. Besides, let $\{\sigma_1, \ldots \sigma_k\}$ be the maximal n-simplices of K, where $\sigma_i = \left(v^i_0, \ldots, v^i_n\right)$ for all i; and let $\{\mu_1, \ldots, \mu_\ell\}$ be the maximal m-simplices of L, where $\mu_j = \left(\, u^j_0, \ldots, u^j_m \,\right)$ for all j. Let us consider a multi-layer feed-forward network $\mathcal{N}_\varphi$ with the following architecture:

(1) An input layer composed of $d_0 = n$ neurons;
(2) a first hidden layer composed of $d_1 = k \cdot (n + 1)$ neurons;
(3) a second hidden layer composed of $d_2 = \ell \cdot (m+1)$ neurons; and
(4) an output layer with $d_3 = m$ neurons.

Then, let $\mathcal{N}_\varphi = f_3 \circ f_2 \circ f_1$ being $f_i(y) = \phi_i(W^{(i)}; y; b_i)$, $i = 1, 2, 3$. Now, the idea is to encode the simplicial complexes involved in the mapping in the hidden layers of the multi-layer feed-forward network. Firstly, a point x in $\mathbb{R}^n$ is transformed into a $k \cdot (n + 1)$ vector. This vector can be seen as the juxtaposition of k vectors of dimension $n + 1$, one for each of the k simplices in K. Each vector of dimension $n + 1$ represents the barycentric coordinates of x with respect to the corresponding simplex. The matrix $W^{(1)} \in \mathcal{M}_{k(n+1)\times n}$ and the bias term b_1 can be obtained from the barycentric coordinates relations as follows. Firstly,

$$W^{(1)} = \begin{pmatrix} W^{(1)}_1 \\ \vdots \\ W^{(1)}_k \end{pmatrix}$$

where $W^{(1)}_i \in \mathcal{M}_{(n+1)\times n}$ is:

$$\begin{pmatrix} v^i_0 & \cdots & v^i_n \\ 1 & \cdots & 1 \end{pmatrix}^{-1} = \left(W^{(1)}_i \quad \middle| \quad B_i\right)$$

being $\{v^i_0, \ldots, v^i_n\}$ the set of vertices of the maximal simplex σ_i of K, and second, $b_1 \in \mathbb{R}^{k(n+1)}$ is:

$$b_1 = \begin{pmatrix} B_1 \\ \vdots \\ B_k \end{pmatrix}.$$

The function ϕ_1 is then defined as:

$$\phi_1(W^{(1)}; y; b_1) = W^{(1)}y + b_1.$$

The matrix of weights $W^{(2)} \in \mathcal{M}_{\ell(m+1)\times k(n+1)}$ between the first and the second hidden layer of $\mathcal{N}_\varphi$ encodes the vertex map φ. The first hidden layer is composed of $k \cdot (n + 1)$ neurons that correspond to the vertices of the maximal simplices of K. The second hidden layer is composed of $\ell \cdot (m+1)$ neurons that correspond to the vertices of the maximal simplices of L. The matrix $W^{(2)}$ is composed of values zeros and ones. On the one hand, an element of $W^{(2)}$ has value 1 if the corresponding vertices in K and L are related by the vertex map φ, and, on the other hand, it has value 0 if they are not related by φ. Then,

$$W^{(2)} = \left(W^{(2)}_{s_1,s_2}\right)$$

where

$$W^{(2)}_{s_1,s_2} = \begin{cases} 1 \text{ if } \varphi(v^i_t) = u^j_r, \\ 0 \text{ otherwise;} \end{cases}$$

being $s_1 = j(r+1)$ and $s_2 = i(t+1)$ for $i = 1,\dots,k$; $j = 1,\dots,\ell$; $t = 0,\dots,n$; and $r = 0,\dots,m$. The bias term b_2 is the null vector. Then, the function ϕ_2 is defined as:

$$\phi_2(W^{(2)}; y; b_2) = W^{(2)}y.$$

The output of the second hidden layer can be seen as the juxtaposition of ℓ vectors of dimension $m+1$, one vector for each simplex in the simplicial complex L. Each of these vectors represents the barycentric coordinates of $\varphi_c(x)$ with respect to the corresponding simplex in L. In the next step, only vectors whose all coordinates are greater than or equal to zero are considered. This condition encodes the simplices of L to which $\varphi_c(x)$ belongs. Then, $\phi_3(W^{(3)}; y; b_3)$ maps the barycentric coordinates of $\varphi_c(x)$ with respect to each maximal simplex of L to which $\varphi_c(x)$ belongs, to the Cartesian coordinates of $\varphi_c(x)$. Specifically,

$$W^{(3)} = \left(W^{(3)}_1 \quad \cdots \quad W^{(3)}_\ell\right) \in \mathcal{M}_{m \times \ell(m+1)},$$

being $W^{(3)}_j = \left(u^j_0 \quad \cdots \quad u^j_m\right)$; and b_3 is the null vector. Finally, ϕ_3 is defined as:

$$\phi_3(W^{(3)}; y; b_3) = \frac{\sum_{j=1}^{\ell} z^j \psi(y^j)}{\sum_{j=1}^{\ell} \psi(y^j)}$$

for $y = \begin{pmatrix} y^1 \\ \vdots \\ y^\ell \end{pmatrix} \in \mathcal{M}^{\ell\cdot(m+1)}$, with $z^j = W^{(3)}_j y^j$ and $\psi(y^j) = 1$ if all the coordinates of y^j are greater than or equal to 0 and $\psi(y^j) = 0$ otherwise.

The particular choice of ϕ_3 and ψ is motivated by the use of the barycentric coordinates. Let us observe that the barycentric coordinates vary with respect to the maximal simplex considered to compute them and that the barycentric coordinates are computed from the coordinates of the vertices of the maximal simplex considered. Besides, maximal simplices can share common vertices. Then, the map ψ is used to determine if a given input is located in a specific simplex. The map ϕ_3 is used to normalize the result in case that a point belongs to more than one simplex. $\square$

Summing up, Theorem 4 establishes that a two-hidden-layer feed-forward network can act equivalently to a simplicial map. The architecture and the specific computation of the parameters of the network are provided in the proof of the theorem.

4. Simplicial approximation theorem extension

In this section, we provide an extension of the Simplicial Approximation Theorem together with an explicit algorithm to compute a simplicial approximation "as close as desired" to a given continuous function $g : |K| \to |L|$ between the underlying spaces of two simplicial complexes K and L. The first observation is that the Simplicial Approximation Theorem (Theorem 3) refers to any continuous function. Nevertheless, continuity is a property of functions between topological spaces, not necessarily metric spaces. The next result introduces the concept of closeness between simplicial approximations and continuous functions.

Proposition 1. *Given $\epsilon > 0$ and a continuous function $g : |K| \to |L|$ between the underlying spaces of two simplicial complexes K and L, there exist $t_1, t_2 > 0$ such that $\varphi_c : |\mathrm{Sd}^{t_1} K| \to |\mathrm{Sd}^{t_2} L|$ is a simplicial approximation of g and $\|g - \varphi_c\| \le \epsilon$.*

Proof. By Theorem 2, there exists t_2 such that $m(\mathrm{Sd}^{t_2} L) \le \epsilon$. Then, by Theorem 3, there exists t_1 such that $\varphi_c : |\mathrm{Sd}^{t_1} K| \to |\mathrm{Sd}^{t_2} L|$ is a simplicial approximation of g:

$$\begin{tikzcd}
|K| \arrow[r, "g"] & |L| \\
K \arrow[u, hook] \arrow[d, "\mathrm{Sd}"] & L \arrow[u, hook] \arrow[d, "\mathrm{Sd}"] \\
\mathrm{Sd}^{t_1} K \arrow[d, hook] & \mathrm{Sd}^{t_2} L \arrow[d, hook] \\
|\mathrm{Sd}^{t_1} K| \arrow[r, "\varphi_c"] & |\mathrm{Sd}^{t_2} L|
\end{tikzcd}$$

Besides, $\|g - \varphi_c\| \le \epsilon$ because $m(\mathrm{Sd}^{t_2} L) \le \epsilon$. $\square$

Algorithm 1: Computing a vertex map that induces a simplicial approximation

---

Input: A continuous function $g : |K| \to |L|$ between the underlying spaces of two simplicial complexes K and L, and an integer t where $\mathrm{Sd}^t\, k$ satisfies the *star condition*: for each $v \in \mathrm{Sd}^t K^{(0)}$ there exists $w \in L^{(0)}$ such that $g(|st(v)|) \subseteq |st(w)|$.

Output: A vertex map φ that induces simplicial approximation φ_c of g.

foreach *vertex $v \in \mathrm{Sd}^t K^{(0)}$* **do**
  Choose $w \in L^{(0)}$ such that $g(|st(v)|) \subseteq |st(w)|$ and define $\varphi(v) := w$.

Algorithm 1 is inspired in the proof of the Simplicial Approximation Theorem given in Edelsbrunner and Harer (2010, p. 56) and computes a vertex map $\varphi : \mathrm{Sd}^t K^{(0)} \to L^{(0)}$ from which we can define the simplicial approximation $\varphi_c : |\mathrm{Sd}^t K| \to |L|$ of a continuous function $g : |K| \to |L|$.

Theorem 5. *Given a continuous function $g : |K| \to |L|$ and $\epsilon > 0$, a two-hidden-layer feed-forward network $\mathcal{N}$ such that $\|g - \mathcal{N}\| \le \epsilon$ can be explicitly defined.*

Proof. By Proposition 1, there exists a simplicial approximation φ_c of g such that $\|g - \varphi_c\| \le \epsilon$, that can be computed using Algorithm 1. Then, by Theorem 4 there exists $\mathcal{N}_\varphi$ such that $\varphi_c = \mathcal{N}_\varphi$. $\square$

5. Universal approximation theorem extension

In the previous sections, we have proved that a continuous function between triangulable spaces can be approximated by using the Simplicial Approximation Theorem. In this section, using the extension of the Simplicial Approximation Theorem given in Proposition 3, we provide a constructive version of the Universal Approximation Theorem that approximates any continuous function (under some specific conditions) arbitrarily close.

Proposition 2. *Let (K, τ) be a finite triangulation of a metric space X. For all $\epsilon > 0$ there exist $t > 0$ and $\gamma > 0$ such that if $m(\mathrm{Sd}^t K) \le \gamma$ then $\tilde{m}_{(\mathrm{Sd}^t K, \tau)}(X) \le \epsilon$.*

Proof. Let us consider $a, b_0 \in |K|$. Then, $a \in \sigma_0$ for some maximal simplex $\sigma_0 \in K$. If b_0 belongs to σ_0 then $\|a - b_0\| \leq \delta(\sigma_0)$ and $d_X(x, y_0) \leq \tilde{\delta}(\sigma_0)$, being $x = \tau^{-1}(a)$ and $y_0 = \tau^{-1}(b_0)$. Otherwise, we repeat the reasoning with $\mathrm{Sd}\, K$. Now, a and b_0 can belong to the same simplex in $\mathrm{Sd}\, K$ or not. If they belong to the same simplex in $\mathrm{Sd}\, K$, write $b_1 = b_0$. If not, take a new point b_1 such that $a, b_1 \in \sigma_1$ and $\sigma_1 \in \mathrm{Sd}\, \sigma_0$. Therefore, $\|a - b_1\| \leq \delta(\sigma_1) \leq \delta(\sigma_0)$. Besides, $d_X(x, y_1) \leq \tilde{\delta}(\sigma_1) \leq \tilde{\delta}(\sigma_0)$ being $x = \tau^{-1}(a)$ and $y_1 = \tau^{-1}(b_1)$ This process can be iterated: $\|a - b_t\| \leq \delta(\sigma_t) \leq \ldots \leq \delta(\sigma_1) \leq \delta(\sigma_0)$ and $d_X(x, y_t) \leq \tilde{\delta}(\sigma_t) \leq \ldots \leq \tilde{\delta}(\sigma_1) \leq \tilde{\delta}(\sigma_0)$, being $x = \tau^{-1}(a)$ and $y_t = \tau^{-1}(b_t)$. By this, we have defined a sequence $\{b_i\}_{i=0}^{t}$ that converges to a. Therefore, given $\epsilon > 0$, there exists t such that $d_X(x, y_t) \leq \epsilon$, being $x = \tau^{-1}(a)$ and $y_t = \tau^{-1}(b_t)$. Let us suppose, without loss of generality, that $\tilde{\delta}(\sigma_t) = \tilde{m}_{(\mathrm{Sd}^t K, \tau)}(X)$. Then, we can consider $\gamma = m(\mathrm{Sd}^t K)$. □

Corollary 1. *Given $\epsilon > 0$ and a finite triangulation (K, τ) of a metric space X, there exists t such that*

$$\tilde{m}_{(\mathrm{Sd}^t K, \tau)}(X) \leq \epsilon.$$

Proof. By Theorem 2 there exists t' such that $m(\mathrm{Sd}^{t'} K) \leq \gamma$. Then, by Proposition 2, there exists t such that $\tilde{m}_{(\mathrm{Sd}^t K, \tau)}(X) \leq \epsilon$. □

Given two continuous functions g_1 and g_2 between two metric spaces X and Y, we denote $\sup\{d_Y(g_1(x), g_2(x)) \mid x \in X\}$ also by $\|g_1 - g_2\|$. Now, given a continuous function g between two finitely triangulable metric spaces X and Y, there exists a simplicial approximation φ_c "arbitrarily close" to g.

Proposition 3. *Let X and Y be two finitely triangulable metric spaces, $g : X \to Y$ a continuous function, and $\epsilon > 0$. Then, there exist two finite triangulations (K, τ_K) and (L, τ_L) of X and Y, respectively, and a simplicial approximation $\varphi_c : |\mathrm{Sd}^{t_1} K| \to |\mathrm{Sd}^{t_2} L|$ such that $\|g - \tilde{\varphi}_c\| \leq \epsilon$ being $\tilde{\varphi}_c = \tau_L^{-1} \circ \varphi_c \circ \tau_K$.*

Proof. First, by Corollary 1, there exists t_2 such that $\tilde{m}_{(\mathrm{Sd}^{t_2} L, \tau_L)}(Y) \leq \epsilon$. Next, by Theorem 3, there exist $t_1 > 0$ and a vertex map $\varphi : (\mathrm{Sd}^{t_1} K)^{(0)} \to (\mathrm{Sd}^{t_2} L)^{(0)}$ such that $\varphi_c : |\mathrm{Sd}^{t_1} K| \to |\mathrm{Sd}^{t_2} L|$ is a simplicial approximation of $\tau_L \circ g \circ \tau_K^{-1}$. Take into account that $|\mathrm{Sd}^{t_1} K| = |K|$ and $|\mathrm{Sd}^{t_2} L| = |L|$. Finally, since $\tilde{m}_{(L, \tau_L)}(Y) \leq \epsilon$ then $\|g - \tilde{\varphi}_c\| \leq \epsilon$. Below, a diagram that schematizes the proof:

$$\begin{tikzcd}
X \arrow[r, "g"] \arrow[d, "\tau_K"] & Y \arrow[d, "\tau_L"] \\
{|K|} \arrow[r, "\varphi_c"] & {|L|} \\
K \arrow[u] \arrow[d, "\mathrm{Sd}"] & L \arrow[u] \arrow[d, "\mathrm{Sd}"] \\
\mathrm{Sd}^{t_1} K & \mathrm{Sd}^{t_2} L \\
(\mathrm{Sd}^{t_1} K)^{(0)} \arrow[u] \arrow[r, "\varphi"] & (\mathrm{Sd}^{t_2} L)^{(0)} \arrow[u]
\end{tikzcd}$$

Finally, we reach the main result of this section: Given a continuous function g between two finitely triangulable spaces X and Y, we can obtain two finite simplicial complexes K and L associated to them, and a multi-layer feed-forward network between the underlying spaces of K and L which "approximates" g.

Theorem 6. *Given a continuous function $g : X \to Y$ between two finitely triangulable metric spaces X and Y and finite triangulations (K, τ_K) and (L, τ_L) of, respectively, X and Y, a two-hidden-layer feed-forward network $\mathcal{N}$ such that $\|g - \tilde{\mathcal{N}}\| \leq \epsilon$, being $\tilde{\mathcal{N}} = \tau_L^{-1} \circ \mathcal{N} \circ \tau_K$, can be explicitly defined.*

Proof. By Proposition 3, there exists a simplicial approximation φ_c such that $\|g - \tilde{\varphi}_c\| \leq \epsilon$. Finally, by Theorem 4, there exists $\mathcal{N}$ such that $\mathcal{N} = \varphi_c$ in all the domains. □

6. Complexity of the architecture of the network

In previous sections, we have provided a constructive approach to build neural networks to approximate continuous functions as close as desired. Now, let us study how the "complexity" of the architecture of the neural network increases in terms of the number of neurons in each hidden layer.

Definition 15. The complexity of a neural network is the maximum of the widths of its hidden layers.

First, let us observe that we can infer an upper bound for the amount of barycentric subdivisions of a simplicial complex needed to reach a specific mesh.

Proposition 4. *Let us consider a finite pure simplicial complex K. Let $\dim(K) = n$ and let $0 < \varepsilon < m(K)$. If*

$$t \geq \frac{\log(m(K)) - \log(\varepsilon)}{\log(n+1) - \log(n)}$$

then $m(\mathrm{Sd}^t(K)) \leq \varepsilon$.

Proof. Let us observe that $m(\mathrm{Sd}^t(K)) \leq m(K) \cdot \left(\frac{n}{n+1}\right)^t$. Then,

$$m(\mathrm{Sd}^t(K)) \leq \varepsilon \Leftarrow m(K) \cdot \left(\frac{n}{n+1}\right)^t \leq \varepsilon \Leftrightarrow t \geq \frac{\log\left(\frac{\varepsilon}{m(K)}\right)}{\log\left(\frac{n}{n+1}\right)}. \quad \square$$

Let us recall that, given two finite pure simplicial complexes K and L with, respectively, k and ℓ maximal simplices, being, respectively, $\dim(K) = n$ and $\dim(L) = m$, the width of the first and the second hidden layer of the neural network $\mathcal{N}$ described in Theorem 4 is, respectively, $k \cdot (n+1)$ and $\ell \cdot (m+1)$. Let us describe how the complexity of $\mathcal{N}$ increases with the iterated applications of the barycentric subdivisions. Let us consider, without loss of generality, that we apply one barycentric subdivision to K. Then, the width of the first hidden layer increases from $k \cdot (n+1)$ to $k \cdot (n+1)! \cdot (n+1)$.

Remark 1. Let us consider a simplicial approximation $\psi_c : |\mathrm{Sd}^{t_1} K| \to |\mathrm{Sd}^{t_2} L|$ of a continuous function, being K and L two finite pure simplicial complexes of dimension n and m, and with k and ℓ maximal simplices, respectively. The complexity of the two-hidden-layer feed-forward network $\mathcal{N}_\psi$ is

$$C(t_1, t_2) = \max\Big\{k\big((n+1)!\big)^{t_1}(n+1),\ \ell\big((m+1)!\big)^{t_2}(m+1)\Big\}.$$

We can relate the modulus of continuity of the input function g and the modulus of continuity of the simplicial approximation φ_c between triangulations of the input spaces. Let us recall the definition of modulus of continuity of a function.

Definition 16. The modulus of continuity of a continuous function $g : X \to Y$ between two metric spaces X and Y is given by:

$$\rho(\delta, g) = \sup\Big\{d_Y(g(x), g(y)) \mid d_X(x, y) \leq \delta\Big\}.$$

In particular we have that:

$$d_X(x, y) \leq \delta \Rightarrow d_Y(g(x), g(y)) \leq \rho(\delta, g).$$

Now, let us study the complexity of the resulting network in terms of the mesh of the triangulation and on the modulus of continuity of the considered function.

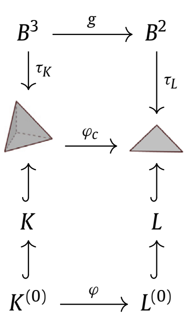
Theorem 7. *Let $g : X \to Y$ be a continuous function between two triangulable metric spaces with modulus of continuity $\rho(\delta, g) \geq 0$. Let us suppose that there exist finite triangulations (K, τ_K) and (L, τ_L) of X and Y, respectively. Then, there exists a two-hidden-layer feed-forward network $\mathcal{N}$ such that $\rho(\delta, \tilde{\mathcal{N}}) \leq 2\rho(\delta, g)$ and $\|g - \tilde{\mathcal{N}}\| \leq \frac{\rho(\delta,g)}{2}$ being $\tilde{\mathcal{N}} = \tau_L^{-1} \circ \mathcal{N} \circ \tau_K$.*

Proof. Let $\epsilon = \frac{\rho(\delta,g)}{2}$. By Theorem 6, there exists a two-hidden-layer feed-forward network $\mathcal{N}$ such that $\|g - \tilde{\mathcal{N}}\| \leq \epsilon$. Consider $x, y \in X$ such that $d_X(x, y) \leq \delta$. then,

$$
\begin{aligned}
& d_Y(\tilde{\mathcal{N}}(x), \tilde{\mathcal{N}}(y)) \\
& \leq d_Y(g(x), \tilde{\mathcal{N}}(x)) + d_Y(g(x), g(y)) + d_Y(g(y), \tilde{\mathcal{N}}(y)) \\
& \leq 2\|g - \tilde{\mathcal{N}}\| + \rho(\delta, g) \leq 2\rho(\delta, g). \quad \square
\end{aligned}
$$

7. Example

In this section, we show a concrete example of a multi-layer feed-forward network approximating a continuous function between two triangulable spaces. The following diagram illustrates the example:



Let us consider the n-dimensional ball $B^n = \{ x \in \mathbb{R}^n \mid 1 \geq \|x\| \}$. Then, a triangulation of B^3 is the simplicial complex K whose maximal simplex is a tetrahedron with set of vertices $K^{(0)} = \{ (0, 0, 0), (1, 0, 0), (0, 1, 0), (0, 0, 1) \}$ and a homeomorphism $\tau_K : B^3 \to |K|$ whose inverse is defined for any point $P \in |K|$ as follows:

$$P' = \tau_K^{-1}(P) = \lambda(P) \cdot \left(P - \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)\right)$$

where $\lambda(P)$ is the quotient $|\overline{OB}|/|\overline{OA}|$ being point $O = (0, 0, 0)$, point A the intersection of the boundary of the tetrahedron with the line that goes through O and P, and point B the intersection of such line with the sphere $S^2 = \{ x \in \mathbb{R}^3 \mid 1 = \|x\| \}$. See Fig. 4 for a similar homeomorphism in the 2-dimensional case.

Besides, let us consider the continuous function $g : B^3 \to B^2$ where $g : B^3 \to B^2$ is the projection given by $g : (P_x, P_y, P_z) \to (P_x, P_y)$ being $P = (P_x, P_y, P_z) \in B^3$. Besides, B^2 can be triangulated by the simplicial complex L whose maximal simplex is the triangle with set of vertices $L^{(0)} = \{(0, 0), (1, 0), (0, 1)\}$ and the homeomorphism $\tau_L : B^2 \to |L|$, whose inverse is

$$P' = \tau_L^{-1}(P) = \lambda(P) \cdot \left(P - \left(\frac{1}{4}, \frac{1}{4}\right)\right)$$

where $P \in |L|$ and $\lambda(P)$ is computed in a similar way than above. See Fig. 4.

Now, let us approximate g with a two-hidden-layer neural network. First, let us observe that a simplicial approximation φ_c of $\tau_L \circ g \circ \tau_K^{-1}$ is given by the vertex set $\varphi : K^{(0)} \to L^{(0)}$ defined as $\varphi((0, 0, 0)) = (0, 0)$, $\varphi((1, 0, 0)) = (1, 0)$, $\varphi((0, 1, 0)) = (0, 1)$, and $\varphi((0, 0, 1)) = (0, 1)$. Once the simplicial approximation is computed, we can determine the specific two-hidden-layer neural network $\mathcal{N}_\varphi$ that acts equivalently to φ_c. Concretely, the architecture of $\mathcal{N}_\varphi$ is composed of an input layer with 3 neurons, a first hidden layer with 4 neurons, a second hidden layer with 3 neurons, and an output layer with 2 neurons. The weights and bias can be computed following the proof of Theorem 4:

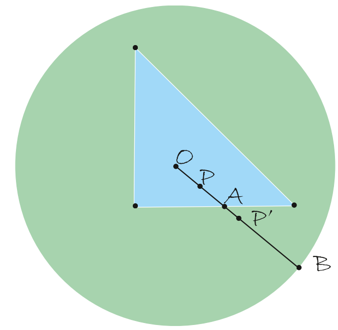


Fig. 4. Geometric visualization of the computation of the parameter λ in the homeomorphism τ_L by which P is mapped to P'.

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} -1 & -1 & -1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Then, $W^{(1)} = \begin{pmatrix} -1 & -1 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ and $b_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$.

$$W^{(2)} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

$$W^{(3)} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

In this straightforward example, the spaces B^3 and B^2 are approximated by just one 3-simplex and one 2-simplex, respectively. Therefore, $\tilde{m}_{(K,\tau_K)}(B^3)$ and $\tilde{m}_{(L,\tau_L)}(B^2)$ are upper bounded by the size of B^3 and B^2, respectively. If we want a better approximation to g, we should apply the barycentric subdivision to K and L, respectively. Doing that, we would obtain six maximal 2-simplices in L, and sixteen maximal 3-simplices in K. Hence, the architecture of the neural network will consist of 64 neurons in the first hidden layer, and 18 neurons in the second hidden layer.

8. Conclusion

In this paper, we have provided an effective method to build a multi-layer feed-forward network which approximates a continuous function between triangulable spaces. The main contribution of the paper is the proof that the weights can be exactly computed without any training process. Although the homeomorphisms between the triangulable spaces and the simplicial complexes can be hard to find and the classic theorem for approximations through neural networks is valid for compact sets and our result is only valid for triangulable spaces, most of the real-world problems are covered by our result and, therefore, approximations to continuous functions through neural networks can effectively be built. Besides, our method can be considered a suitable and powerful tool for the approximation of continuous functions on triangulable spaces. Two of the main advantages of the proposed method are: (1) knowing a priori how many hidden neurons are needed to reach the desired accuracy; and (2) no need for a training process.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Ayala, R., Domínguez, E., & Quintero, A. (2002). Elementos de la teoría de homología clásica. In *Ciencias (universidad de sevilla)*, Secretariado de Publicaciones, Universidad de Sevilla, URL: https://books.google.es/books?id=CAOjRFAMJFUC.

Boissonnat, J., Chazal, F., & Yvinec, M. (2018). Geometric and topological inference. In *Cambridge texts in applied mathematics*, Cambridge University Press, URL: https://books.google.es/books?id=0rBoDwAAQBAJ.

Cohen, N., Sharir, O., & Shashua, A. (2016). On the expressive power of deep learning: A tensor analysis. In V. Feldman, A. Rakhlin, & O. Shamir (Eds.), *Proceedings of the 29th conference on learning theory, COLT 2016, Vol. 49* (pp. 698–728). New York, USA: JMLR.org, URL: http://proceedings.mlr.press/v49/cohen16.html.

Cohen, N., & Shashua, A. (2016). Convolutional rectifier networks as generalized tensor decompositions. In M. Balcan, & K. Q. Weinberger (Eds.), *Proceedings of the 33nd international conference on machine learning, ICML 2016, Vol. 48* (pp. 955–963). New York City, NY, USA: JMLR.org, URL: http://proceedings.mlr.press/v48/cohenb16.html.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control Signal and Systems*, *2*(4), 303–314. http://dx.doi.org/10.1007/BF02551274.

Delalleau, O., & Bengio, Y. (2011). Shallow vs. Deep sum-product networks. In Shawe-Taylor, J., and Zemel, R.S., and Bartlett, P.L., and Pereira, F.C.N., and Weinberger, K.Q., (Eds.) *Advances in neural information processing systems 24: 25th Annual conference on neural information processing Systems 2011. Proceedings of a meeting held* 12-14 December 2011, (pp. 666–674), Granada, Spain, URL: http://papers.nips.cc/paper/4350-shallow-vs-deep-sum-product-networks.

Edelsbrunner, H., & Harer, J. (2010). *Computational topology - an introduction*. American Mathematical Society.

Geoghegan, R. (2010). Topological methods in group theory. In *Graduate texts in mathematics*, Springer New York, URL: https://books.google.ch/books?id=MEKZcQAACAAJ.

Guliyev, N. J., & Ismailov, V. E. (2018). On the approximation by single hidden layer feedforward neural networks with fixed weights. *Neural Networks*, *98*, 296–304. http://dx.doi.org/10.1016/j.neunet.2017.12.007, URL: http://www.sciencedirect.com/science/article/pii/S0893608017302927.

Hanin, B., & Sellke, M. (2017). Approximating continuous functions by relu nets of minimal width. arXiv:arXiv:1710.11278.

Hatcher, A. (2002). *Algebraic topology*. Cambridge: Cambridge University Press.

Haykin, S. (1999). *Neural networks: A comprehensive foundation*. Prentice Hall.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, *4*(2), 251–257. http://dx.doi.org/10.1016/0893-6080(91)90009-T.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*, 356–366.

Kileel, J., Trager, M., & Bruna, J. (2019). On the expressive power of deep polynomial neural networks. In Wallach, H.M., and Larochelle, H., and Beygelzimer, A., and d'Alché-Buc, F., and Fox, E.B., and Garnett, R., (Eds.) *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019*, 8-14 December 2019, (pp.10310–10319). Vancouver, BC, Canada, URL: http://papers.nips.cc/paper/9219-on-the-expressive-power-of-deep-polynomial-neural-networks.

Liang, S., & Srikant, R. (2016). Why deep neural networks? CoRR abs/1610.04161 URL: arXiv:1610.04161.

Lu, Z., Pu, H., Wang, F., Hu, Z., & Wang, L. (2017). The expressive power of neural networks: A view from the width. In Guyon, I., and von Luxburg, U., and Bengio, S., and Wallach, H.M., and Fergus, R., and Vishwanathan, S.V.N., and Garnett, R., (Eds.) *Advances in neural information processing systems 30: annual conference on neural information processing systems 2017* 4-9 December 2017, (pp. 6232–6240). Long Beach, CA, USA, URL: http://papers.nips.cc/paper/7203-the-expressive-power-of-neural-networks-a-view-from-the-width.

Martens, J., & Medabalimi, V. (2014). On the expressive efficiency of sum product networks. CoRR abs/1411.7717 URL: http://dblp.uni-trier.de/db/journals/corr/corr1411.html#MartensM14.

Munkres, J. R. (1984). *Elements of algebraic topology*. Addison-Wesley.

Nguyen, Q., Mukkamala, M. C., & Hein, M. (2018). Neural networks should be wide enough to learn disconnected decision regions. CoRR abs/1803.00094 URL: arXiv:1803.00094.

Poon, H., & Domingos, P. (2012). Sum-product networks: A new deep architecture. arXiv:1202.3732.

Safran, I., & Shamir, O. (2017). Depth-width tradeoffs in approximating natural functions with neural networks. In *Proceedings of the 34th international conference on machine learning, Vol. 70* (pp. 2979–2987). Sydney, Australia: PMLR, International Convention Centre, URL: http://proceedings.mlr.press/v70/safran17a.html.

Sun, S., Chen, W., Wang, L., Liu, X., & Liu, T. (2016). On the depth of deep neural networks: A theoretical view. In D. Schuurmans, & M. P. Wellman (Eds.), *Proceedings of the thirtieth AAAI conference on artificial intelligence* (pp. 2066–2072). Phoenix, Arizona, USA: AAAI Press, URL: http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12073.

Telgarsky, M. (2016). Benefits of depth in neural networks. CoRR abs/1602.04485 URL: arXiv:1602.04485.