mathematics

MDPI

Article

Simplicial-Map Neural Networks Robust to Adversarial Examples

Eduardo Paluzo-Hidalgo [1,*,†], Rocio Gonzalez-Diaz [1,†,‡], Miguel A. Gutiérrez-Naranjo [2,†,‡] and Jónathan Heras [3,‡]

[1] Department of Applied Mathematics I, University of Seville, 41012 Seville, Spain; rogodi@us.es
[2] Department of Computer Sciences and Artificial Intelligence, University of Seville, 41012 Seville, Spain; magutier@us.es
[3] Department of Mathematics and Computer Sciences, University of La Rioja, 26006 Logroño, Spain; jonathan.heras@unirioja.es
* Correspondence: epaluzo@us.es
† These authors are partially supported by MICINN, FEDER/UE under grant PID2019-107339GB-100.
‡ These authors contributed equally to this work.

Abstract: Broadly speaking, an adversarial example against a classification model occurs when a small perturbation on an input data point produces a change on the output label assigned by the model. Such adversarial examples represent a weakness for the safety of neural network applications, and many different solutions have been proposed for minimizing their effects. In this paper, we propose a new approach by means of a family of neural networks called *simplicial-map neural networks* constructed from an Algebraic Topology perspective. Our proposal is based on three main ideas. Firstly, given a classification problem, both the input dataset and its set of one-hot labels will be endowed with simplicial complex structures, and a simplicial map between such complexes will be defined. Secondly, a neural network characterizing the classification problem will be built from such a simplicial map. Finally, by considering barycentric subdivisions of the simplicial complexes, a decision boundary will be computed to make the neural network robust to adversarial attacks of a given size.

Keywords: algebraic topology; neural network; adversarial examples

Citation: Paluzo-Hidalgo, E.; Gonzalez-Diaz, R.; Gutiérrez-Naranjo, M.A.; Heras, J. Simplicial-Map Neural Networks Robust to Adversarial Examples. *Mathematics* **2021**, *9*, 169. https://doi.org/10.3390/math9020169

Received: 11 December 2020
Accepted: 12 January 2021
Published: 15 January 2021

1. Introduction

Adversarial examples are currently one of the main problems for the robustness of neural networks applications [1]. Broadly speaking, an adversarial example against a classification model occurs when a small perturbation on an input data point produces a change on its classification. Adversarial examples are usually associated with computer vision tasks [2]. In this context, *small* generally refers to changes that are not appreciable by human perception. Recently, several studies have shown that adversarial examples also appear in other contexts such as natural language processing [3], multivariate time series [4] or recommendation systems [5]. Therefore, the study of adversarial examples is of an undoubted importance for building reliable models, and also leads us to wonder about the mechanisms of our brain and the differences between artificial and natural classification processes.

Since the discovery of adversarial examples as a weakness for the safety of neural network models in real-world problems, many attacks and defenses have been proposed [6], each of which builds on the other. One of the most popular approaches to study models' robustness and adversarial examples is based on the concept of *margin*. From a geometrical point of view, data are points in an d-dimensional metric space and a *classifier* splits such a metric space into regions. In a classification problem, each region is associated with a *label* and all the points in such a region are classified with the corresponding label. Roughly

speaking, the margin of the model is the minimum distance between the training data and the *decision boundary* (i.e., the set of regions' boundaries). In the literature, there are many approaches that try to maximize such a margin (see, for instance, [7]). The concept of margin in Neural Network is strongly influenced by its use in Support Vector Machines (SVMs) [8]. In such way, in [9], the final softmax layer of the neural network is replaced with a linear SVM. In [10], a way of reducing empirical margin errors was proposed, and in [11], the discriminability of Deep Neural Networks (DNNs) features is enhanced via an ensemble strategy.

In this paper, we explore this idea of margin between regions associated with labels from a novel point of view. To the best of our knowledge, this is the first time where adversarial examples are studied with techniques from Algebraic Topology. Our approach can be summarized as follows. The starting point is a classification problem where the data are d-dimensional vectors which are mapped onto a set of k labels with a one-hot representation. From a topological point of view, such set of instances can be seen as the vertices of a simplicial complex embedded in a bounded polytope in $\mathbb{R}^d$ (details are given below), and the set of k one-hot labels can be endowed with the structure of a k-dimensional simplex (in fact, $k+1$ labels are considered since we add an *unknown* label to the set of one-hot labels). In this way, a simplicial map between both topological structures arises in a natural manner, since each vertex in the simplicial complex corresponds to an instance of the dataset, and it is mapped onto the vertex of the simplex that represents the corresponding label. The second step is to apply the extended Simplicial Approximation Theorem [12] that allows us to provide a constructive proof of the Universal Approximation Theorem obtaining a a neural network that classifies correctly all the instances of the dataset. As shown in [12], all the weights of such a neural network can be computed directly from the simplicial complexes without any kind of training processes. Finally, by considering these ideas together with a subdivision process of the simplices, a decision boundary for the classification problem will be computed to make the neural network robust to adversarial attacks of a given size, since the mesh of a simplicial complex can be bounded by the number of subdivisions and the neural network obtained is based on the simplices used in the simplicial map.

Regarding other approaches to these ideas found in the literature, in [13], the authors proved the existence of a two-hidden-layer neural network which can approximate any continuous multivariable function with arbitrary precision, and, in [14], they provided a constructive method through a numerical analysis approach. Therefore, such papers can be seen as alternative constructive proofs to the Universal Approximation Theorem where no adversarial examples on classification problems were considered. A related approach that uses simplicial complexes to feed a neural network, is the concept of *Simplicial Neural Network (SNN)*, provided in [15], that consists of a generalization of *Graph Neural Networks (GNNs)* with the property that compared to GNNs, SNNs exploit higher order relationships between the input data due to representing the data using simplicial complexes. Let us observe that, although having a similar name, our approach has a totally different goal.

The paper is organized as follows. In Section 2, all the basic concepts needed to understand the rest of the paper are presented. In Section 3, we introduce the concept of simplicial-map neural networks. Their use to build neural networks for classification tasks robust to adversarial attacks of a given size is presented in Section 4. The paper ends with conclusions and future works listed in Section 5.

2. Background

In this section, some of the preliminary concepts from Algebraic Topology and Neural Networks are recalled. Several useful references for this section are [16–19]. Let us notice that, in order to provide a bridge between Algebraic Topology and Neural Network, some concepts need to be reinterpreted.

Firstly, let us state some basic notation. Given two integers $j \leq m$, let $[\![j, m]\!] := \{i \in \mathbb{Z} : j \leq i \leq m\}$. Hereafter, let $k > 0$ be an integer and let $e_0^k := (0, \ldots, 0)$ be the origin of the

Euclidean space $\mathbb{R}^k$. Let a *one-hot vector* of length k be denoted as $e_i^k = (0, \overset{i-1}{\cdots}, 0, 1, 0, \overset{k-i}{\cdots}, 0)$ with $i \in [\![1, k]\!]$. Let $\mathbb{E}^k := \{e_i^k : i \in [\![1, k]\!]\}$ be the set of all the one-hot vectors of length k. Let us observe that $\mathbb{E}^{k+1} = \{e_i^k \times 0 : i \in [\![1, k]\!]\} \cup \{e_0^k \times 1\}$ where $e_i^k \times 0 := (0, \overset{i-1}{\cdots}, 0, 1, 0, \overset{k-i}{\cdots}, 0, 0) = e_i^{k+1}$ for $i \in [\![1, k]\!]$ and $e_0^k \times 1 := (0, \overset{k}{\cdots}, 0, 1) = e_{k+1}^{k+1}$.

Now, we recall different fundamental structures such as polytopes and simplicial complexes. Convex polytopes can be seen as a generalization in any dimension of the notion of polygons.

Definition 1. *The convex hull of a set $S \subset \mathbb{R}^d$, denoted by $\mathrm{conv}(S)$, is the smallest convex set containing S. A convex polytope $\mathcal{P}$ in $\mathbb{R}^d$ is the convex hull of a finite set of points. Besides, the set of vertices of a convex polytope $\mathcal{P}$ is the minimum set $V_\mathcal{P}$ of points in $\mathcal{P}$ such that $\mathcal{P} = \mathrm{conv}(V_\mathcal{P})$.*

Accordingly, a convex polytope $\mathcal{P} = \mathrm{conv}(S)$ is a closed bounded subset of $\mathbb{R}^d$, and the set of vertices $V_\mathcal{P}$ of a convex polytope $\mathcal{P}$ always exists and it is unique. A particular case of convex polytopes are simplices. Geometrically, a simplex is a generalization of a triangle to any dimension. For example, a 0-simplex is a point, a 1-simplex is a line segment, a 2-simplex is a triangle, a 3-simplex is a tetrahedron, and so on. In this paper, all the considered simplicial complexes have their vertices in the Euclidean space $\mathbb{R}^d$. Nevertheless, simplicial complexes can be defined abstractly.

Definition 2. *Let us consider a finite set V whose elements will be called vertices. A simplicial complex K consists of a finite collection of nonempty subsets (called simplices) of V such that:*

1. *Any subset of V with exactly one point of V is a simplex of K called 0-simplex or vertex.*
2. *Any nonempty subset of a simplex σ is a simplex, called a face of σ.*

A simplex σ with exactly $k+1$ points is called a k-simplex. We also say that the dimension of σ is k and write $\dim \sigma = k$. A maximal simplex of K is a simplex that is not face of any other simplex in K. The dimension of K is denoted by $\dim K$ and it is the maximum dimension of its maximal simplices. The set of vertices of a simplicial complex K will be denoted by $K^{(0)}$. For a vertex v of V, the star of v is the set of simplices having v as a face and it is denoted by $\mathrm{st}\, v$. A simplicial complex K is pure if all its maximal simplices have the same dimension.

Let us consider a simplicial complex K whose vertices are in $\mathbb{R}^d$. If a k-simplex σ of K satisfies that it is a set of affinely independent points, then its *realization* $|\sigma|$ is the convex polytope $|\sigma| = \mathrm{conv}(\sigma)$, which is the convex hull of its $k+1$ vertices. If all the simplices of K have a realization in $\mathbb{R}^d$ satisfying that the intersection of two realizations is the realization of a simplex of K, then the union of their realizations is a subspace of $\mathbb{R}^d$ denoted by $|K|$ and called the embedding of K in $\mathbb{R}^d$.

Next, the definition of triangulation of a convex polytope is recalled.

Definition 3. *A triangulation of a convex polytope $\mathcal{P}$ is a simplicial complex K such that $|K| = \mathcal{P}$.*

Let us recall that given a set $S = \{p_1, \ldots, p_n\}$ of points in $\mathbb{R}^d$, its barycenter, denoted by $\mathrm{bar}\, S$, is $\mathrm{bar}\, S := \frac{1}{n} \sum_{i \in [\![1,n]\!]} p_i \in \mathbb{R}^d$. In particular, $\mathrm{bar}\{p\} = p$ for $p \in \mathbb{R}^d$. The barycentric subdivision of a simplicial complex will be the main tool to *refine* the neural network in the next sections and consists of getting a new simplicial complex by *splitting* the simplices in a standard way (see Figure 1). The t-th iteration of the barycentric subdivision of a simplicial complex K will be denoted by $\mathrm{Sd}^t K$ being $\mathrm{Sd}^0 K := K$. Next definition provides a formalization of this idea.

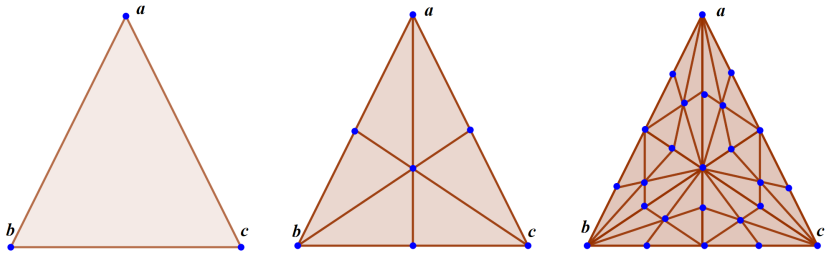


Figure 1. Example of a barycentric subdivision. Let $V = \{a, b, c\}$ be the set of the three vertices (in blue) of the triangle depicted on the left. Let $K = \{\{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}$. From left to right: $|K|$, $|\operatorname{Sd} K|$, and $|\operatorname{Sd}^2 K|$ are shown.

Definition 4. *Let K be a simplicial complex with vertices in $\mathbb{R}^d$. The barycentric subdivision $\operatorname{Sd} K$ is the simplicial complex defined as follows. The set $(\operatorname{Sd} K)^{(0)}$ of vertices of $\operatorname{Sd} K$ is the set of barycenters of all the simplices of K. The simplices of $\operatorname{Sd} K$ are the finite nonempty collections of $(\operatorname{Sd} K)^{(0)}$ which are totally ordered by the face relation in K. That is, any k-simplex σ of $\operatorname{Sd} K$ can be written as an ordered set $\{w_0, \ldots, w_k\}$ such that $w_i = \operatorname{bar} \mu_i$ being μ_i a face of $\mu_j \in K$ for $i, j \in [\![0, k]\!]$ and $i < j$. In particular, if σ is maximal then there exists a k-simplex $\{u_0, \ldots, u_k\} \in K$ satisfying that $w_i = \operatorname{bar}\{u_0, \ldots, u_i\}$ for $i \in [\![0, k]\!]$.*

Let us recall the definition of the Voronoi diagram of a set of points.

Definition 5. *Let $S = \{p_1, \ldots, p_n\}$ be a set of points in $\mathbb{R}^d$. The Voronoi cell $\mathcal{V}(p_i, S)$ is defined as:*

$$\mathcal{V}(p_i, S) := \{x \in \mathbb{R}^d : ||x - p_i|| \leq ||x - p_j||, \forall p_j \in S\}.$$

Then, the Voronoi diagram of S, denoted as $\mathcal{V}(S)$, is the set of Voronoi cells:

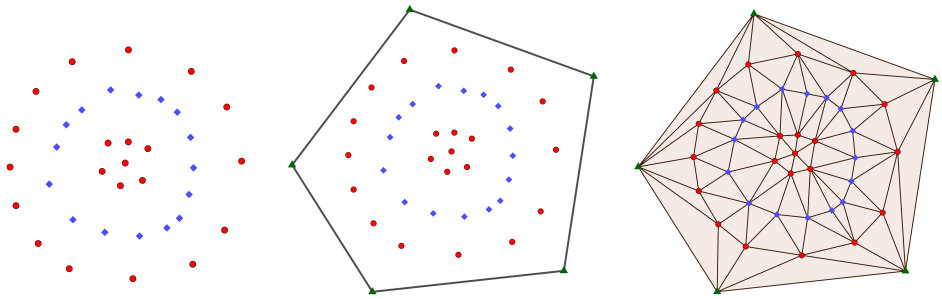
$$\mathcal{V}(S) := \{\mathcal{V}(p_1, S), \ldots, \mathcal{V}(p_n, S)\}.$$

From the Voronoi diagram $\mathcal{V}(S)$, a particular simplicial complex, called the Delaunay complex of S and denoted as $\mathscr{D}(S)$, can be constructed. Both structured can be computed in time $\Theta(n \log n + n^{\lceil \frac{d}{2} \rceil})$ (see [17], Chapter 4).

Definition 6. *Given a finite set of points $S = \{p_1, \ldots, p_n\}$ in $\mathbb{R}^d$ and its Voronoi diagram $\mathcal{V}(S) = \{\mathcal{V}(p_1, S), \ldots, \mathcal{V}(ps_n, S)\}$, the Delaunay complex of S can be defined as:*

$$\mathscr{D}(S) := \{\varsigma \subseteq S : \cap_{p \in \varsigma} \mathcal{V}(p, S) \neq \emptyset\}.$$

The Delaunay complex is a well-defined concept in the sense that $\mathscr{D}(S)$ is always a simplicial complex [17]. Usually, a finite set of points $S \subset \mathbb{R}^d$ is said to be in general position when any subset of S with size at most $d + 1$ is a set of affinely independent points. When the set of points $S \subset \mathbb{R}^d$ is in general position, then the embedding of the Delaunay complex $\mathscr{D}(S)$ in $\mathbb{R}^d$ is a triangulation of $\mathcal{P} = \operatorname{conv}(S)$. In Figure 2, an example of the computation of a triangulation of a convex polytope $\mathcal{P}$, being the Delaunay complex of the set of vertices of $\mathcal{P}$ together with a labelled dataset lying in the interior of $\mathcal{P}$, is provided.



(**a**) Labelled dataset D (**b**) Convex polytope $\mathcal{P}$ (**c**) Simplicial complex K

Figure 2. Given a labelled dataset D, a convex polytope $\mathcal{P}$ containing D can be computed. Then, the simplicial complex K can be obtained using the Delaunay triangulation of all the points of D and the vertices of $\mathcal{P}$.

Let us see now how to define maps between simplicial complexes.

Definition 7. *Given two simplicial complexes K and L, a vertex map $\varphi^{(0)} : K^{(0)} \to L^{(0)}$ is a function from the vertices of K to the vertices of L such that for any simplex $\sigma \in K$, the set*

$$\varphi(\sigma) := \{v \in L^{(0)} : \exists\, u \in \sigma,\ \varphi^{(0)}(u) = v\}$$

is a simplex of L.

Let us observe that $\varphi(u) = \varphi^{(0)}(u)$ if $u \in K^{(0)}$ and the composition of vertex maps is a vertex map. Let us see now that a vertex map $\varphi^{(0)} : K^{(0)} \to L^{(0)}$ can always be extended to a continuous function $\varphi^c : |K| \to |L|$ satisfying that if $x = \operatorname{bar}\sigma$ then $\varphi^c(x) = \operatorname{bar}\varphi(\sigma)$.

Definition 8. *The simplicial map $\varphi^c : |K| \to |L|$ induced by the vertex map $\varphi^{(0)} : K^{(0)} \to L^{(0)}$ is a continuous function defined as follows. Let $x \in |K|$. Then,*

$$\varphi^c(x) := \sum_{i \in [\![0,k]\!]} \lambda_i \varphi^{(0)}(u_i),$$

being $\lambda_i \geq 0$, for all $i \in [\![0,k]\!]$, such that

$$\sum_{i \in [\![0,k]\!]} \lambda_i = 1 \text{ and } x = \sum_{i \in [\![0,k]\!]} \lambda_i u_i,$$

where $\sigma = \{u_0, \ldots, u_k\}$ is a simplex of K such that $x \in |\sigma|$.

Next, we recall one of the key ideas in this paper. Simplicial maps can be used to approximate continuous functions as closed as desired.

Definition 9. *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 $\varphi^{(0)} : K^{(0)} \to L^{(0)}$ is a simplicial approximation of g if*

$$g(|\operatorname{st} v|) \subseteq |\operatorname{st} \varphi(v)|$$

for each vertex v of K.

Let us notice that $|\operatorname{st} x|$ is thought here as an open set of points. That is, $|\operatorname{st} x| := \cup_{\sigma \in \operatorname{st} x} \operatorname{int}\sigma$.

Theorem 1. *Simplicial Approximation Theorem ([20], p. 56) If $g : |K| \to |L|$ is a continuous function between the underlying spaces of two simplicial complexes K and L, then there is a sufficiently large integer $t > 0$ such that $\varphi^c : |Sd^t K| \to |L|$ is a simplicial approximation of g.*

In Figure 3, an example of a simplicial approximation is provided. Theorem 1 was extended in [12] by introducing a bound to the distance between the continuous function and its simplicial approximation.

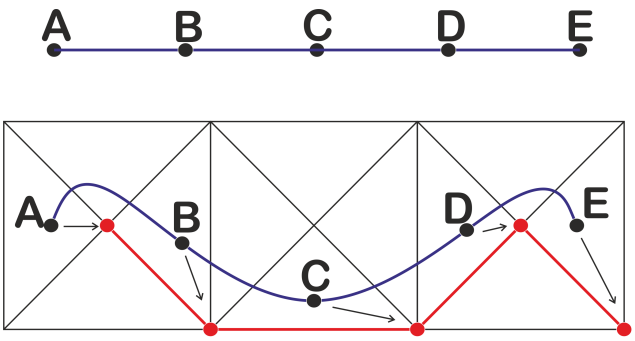


Figure 3. On the top, we can see a 1-simplex with two iterated applications of the barycentric subdivision. On the bottom, a continuous function was applied to the straight line and a simplicial approximation (in red) is provided. The star condition is satisfied and no more barycentric subdivisions are needed.

Proposition 1 (Simplicial Approximation Theorem Extension [12]). *Given $\epsilon > 0$ and a continuous function $g : |K| \to |L|$ between the underlying spaces of two simplicial complexes K and L, there exists $s, t > 0$ such that $\varphi^c : |\mathrm{Sd}^s K| \to |\mathrm{Sd}^t L|$ is a simplicial approximation of g and $||g - \varphi^c|| \leq \epsilon$.*

Once concepts from Algebraic Topology have been stated, let us provide the definition of neural network, and a connection between these two fields using results from [12].

Definition 10 (adapted from [19]). *Given $d, k > 0$, a multi-layer feed-forward network defined between spaces $X \subseteq \mathbb{R}^d$ and $Y \subseteq \mathbb{R}^k$ is a function $\mathcal{N} : X \to Y$ composed by $m + 1$ functions:*

$$\mathcal{N} = f_{m+1} \circ f_m \circ \cdots \circ f_1$$

where the integer $m > 0$ is the number of hidden layers and, for $i \in [\![1, m+1]\!]$, the function $f_i : X_{i-1} \to X_i$ is defined as

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

where $X_0 = X$, $X_{m+1} = Y$, and $X_i \subseteq \mathbb{R}^{d_i}$ for $i \in [\![1, m]\!]$; $d_0 = d$, $d_{m+1} = k$, and $d_i > 0$ being an integer for $i \in [\![1, m]\!]$ (called the width of the i-th hidden layer); $W^{(i)} \in \mathcal{M}_{d_{i-1} \times d_i}$ being a real-valued $d_{i-1} \times d_i$ matrix (called the matrix of weights of $\mathcal{N}$); b_i being a point in $\mathbb{R}^{d_i}$ (called the bias term); and ϕ_i being a function (called the activation function).

In the literature, many other definitions of neural networks are available. The field is continuously adding new ideas and there is not a general definition which covers all the possible approaches, but many of the problems where neural networks are applied are based on the idea of finding a set of weights and bias where the remaining features of the neural network (number of hidden layers, their dimension, and activation functions) are settled at the beginning of the problem. As usual, such set of features of the neural network beyond the weights and the bias, will be called the *architecture* of the neural network. A constructive method for approximating multidimensional functions with neural networks was provided in [12]. Such networks have two hidden layers and the weights are not obtained by a training method, but they are determined by a given simplicial map.

Theorem 2 (Theorem 4 of [12]). *Let us consider a simplicial map $\varphi^c : |K| \to |L|$ between the embedding of two finite pure simplicial complexes K and L of dimension d and k, respectively. Then a two-hidden-layer feed-forward network $\mathcal{N}_\varphi : |K| \to |L|$ such that $\mathcal{N}_\varphi(x) = \varphi^c(x)$ for all $x \in |K|$ can be explicitly defined.*

The construction of the neural network given in [12] to prove Theorem 2 gives rise to the concept of simplicial-map neural network introduced in the next section.

3. Simplicial-Map Neural Networks

The explicit construction of the neural network given in [12] is the main tool used in this paper for computing neural networks robust to adversarial attacks. Such a concrete construction is called *simplicial-map neural network*.

Definition 11. *Let K and L be two finite pure simplicial complexes of dimension d and k, respectively. Let us consider the simplicial map $\varphi^c : |K| \to |L|$ induced by a vertex map $\varphi^{(0)} : K^{(0)} \to L^{(0)}$. Let $\{\sigma_1, \dots \sigma_n\}$ be the maximal simplices of K, where $\sigma_s = \{u_0^s, \dots, u_d^s\}$ and $u_h^s \in \mathbb{R}^d$ for $s \in [\![1, n]\!]$ and $h \in [\![0, d]\!]$. Let $\{\mu_1, \dots, \mu_m\}$ be the maximal simplices of L, where $\mu_j = \{v_0^j, \dots, v_k^j\}$ and $v_h^j \in \mathbb{R}^k$ for $j \in [\![1, m]\!]$ and $h \in [\![0, k]\!]$. The simplicial-map neural network induced by φ^c is a two-hidden-layer feed-forward neural network denoted by $\mathcal{N}_\varphi$ with the following architecture:*

- *an input layer composed of $d_0 = d$ neurons;*
- *a first hidden layer composed of $d_1 = n(d+1)$ neurons;*
- *a second hidden layer composed of $d_2 = m(k+1)$ neurons; and*
- *an output layer with $d_3 = k$ neurons.*

Then, $\mathcal{N}_\varphi = f_3 \circ f_2 \circ f_1$ being

$$f_i(y) = \phi_i(W^{(i)}; y; b_i), \text{ for } i \in [\![1, 3]\!].$$

Firstly, $W^{(1)} = \begin{pmatrix} W_1^{(1)} \\ \vdots \\ W_n^{(1)} \end{pmatrix} \in \mathcal{M}_{n(d+1)\times d}$ *being*

$$\begin{pmatrix} W_i^{(1)} & | & B_i \end{pmatrix} = \begin{pmatrix} u_0^s & \cdots & u_d^s \\ 1 & \cdots & 1 \end{pmatrix}^{-1} \in \mathcal{M}_{(d+1)\times(d+1)}$$

where $W_i^{(1)} \in \mathcal{M}_{(d+1)\times d}$ *and* $B_i \in \mathbb{R}^{d+1}$. *The bias term* $b_1 \in \mathbb{R}^{n(d+1)}$ *is* $b_1 = \begin{pmatrix} B_1 \\ \vdots \\ B_n \end{pmatrix}$ *and the function ϕ_1 is then defined as:*

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

Secondly, $W^{(2)} = \left(W_{h,\ell}^{(2)}\right) \in \mathcal{M}_{m(k+1)\times n(d+1)}$ *where*

$$W_{h,\ell}^{(2)} := \begin{cases} 1 & \text{if } \varphi^{(0)}(u_t^s) = v_r^j, \\ 0 & \text{otherwise;} \end{cases}$$

being $h = j(r+1)$ and $\ell = s(t+1)$ for $s \in [\![1, n]\!]$; $j \in [\![1, m]\!]$; $t \in [\![0, d]\!]$; and $r \in [\![0, k]\!]$. The bias term $b_2 \in \mathbb{R}^{m(k+1)}$ is null and the function ϕ_2 is defined as:

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

Thirdly, $W^{(3)} = \begin{pmatrix} W_1^{(3)} & \cdots & W_m^{(3)} \end{pmatrix} \in \mathcal{M}_{k\times m(k+1)}$ *being*

$$W_j^{(3)} := \begin{pmatrix} v_0^j & \cdots & v_k^j \end{pmatrix} \text{ for } j \in [\![1, m]\!],$$

the bias term b_3 is null, and ϕ_3 is defined as:

$$\phi_3(W^{(3)}; y; b_3) := \frac{\sum_{j\in[\![1,\ell]\!]} z^j \psi(y^j)}{\sum_{j\in[\![1,\ell]\!]} \psi(y^j)}$$

being $z^j := W^{(3)}_j y^j$ *for* $y = \begin{pmatrix} y^1 \\ \vdots \\ y^m \end{pmatrix} \in \mathcal{M}^{m\cdot(k+1)}$ *and*

$$\psi(y^j) := \begin{cases} 1 & \text{if all the coordinates of } y^j \text{ are } \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

In [12], it is proven that $\mathcal{N}_\varphi(x)$ and $\varphi^c(x)$ coincide for all $x \in |K|$.

Proposition 2 ([12]). *Let K and L be two finite pure simplicial complexes of dimension $d > 0$ and $k > 0$, respectively. Let us consider the simplicial map $\varphi^c : |K| \to |L|$ induced by a vertex map $\varphi^{(0)} : K^{(0)} \to L^{(0)}$. Then, the simplicial-map neural network $\mathcal{N}_\varphi : |K| \to |L|$ induced by the simplicial map φ^c satisfies that $\mathcal{N}_\varphi(x) = \varphi^c(x)$ for all $x \in |K|$.*

4. Classification with Simplicial-Map Neural Networks

In this section, simplicial-map neural networks are considered as tools for classification tasks and for the study of adversarial examples. As usual, the classification problem will consist of finding a set of weights adapted to a labelled dataset given a fixed architecture.

Definition 12. *Let $n, d, k > 0$ be integers. A labelled dataset D is a finite set of pairs*

$$D = \{(p_j, \ell_j) : j \in [\![1, n]\!],\ \ p_j \in \mathbb{R}^d,\ \ell_j \in \mathbb{E}^k\}$$

where, for $j, h \in [\![1, n]\!]$, $p_j \neq p_h$ if $j \neq h$, and ℓ_j represents a one-hot vector. We say that ℓ_j is the label of p_j or, equivalently, that p_j belongs to the class ℓ_j. Besides, we will denote by D_P the ordered set of points $\langle p_j \rangle_j$.

The concept of supervised classification problem for neural networks can be defined as follows.

Definition 13. *Given a labelled dataset $D \subset \mathbb{R}^d \times \mathbb{E}^k$, an integer $m > 0$, and activation functions ϕ_i for $i \in [\![1, m]\!]$, a supervised classification problem consists of looking for the weights $W^{(i)}$ and bias terms b_i for $i \in [\![1, m]\!]$, such that the associated neural network $\mathcal{N} : X \to Y$, with $X \subseteq \mathbb{R}^d$, $Y \subseteq \mathbb{R}^k$ and $D \subseteq X \times Y$, satisfies:*

- *$\mathcal{N}(p) = \ell$ for all $(p, \ell) \in D$.*
- *$\mathcal{N}$ maps $x \in X$ to a vector of scores $\mathcal{N}(x) = (y_1, \ldots, y_k) \in Y$ such that $y_i \in [0, 1]$ for $i \in [\![1, n]\!]$ and $\sum_{i\in[\![1,n]\!]} y_i = 1$.*

If such a neural network $\mathcal{N}$ exists, we will say that $\mathcal{N}$ characterizes D, or, equivalently, that $\mathcal{N}$ correctly classifies D.

Let us remark that the success of a classification model as a neural network is not usually measured on the correct classification on the input dataset, but on the correct classification of unseen examples, (that is, pairs not in D), collected in a *test* set. In this paper, we chose such a restrictive definition since we are more interested in dealing with the problem of the robustness of neural networks against adversarial attacks than in the problem of overfitting. Besides, let us observe that, as usual, the scores can be interpreted as a probability distribution over the labels.

Remark 1. *It is known that some functions like the logistic sigmoid, the softmax or the softplus satisfy the properties of a probability distribution and they are broadly applied in deep learning*

models. Our function also behaves like a probability distribution which is adequate for multiclassification tasks.

Next, we provide the definition of some of the main concepts in this paper, the confidence set $T_{\mathcal{N}}$, the classified set $C_{\mathcal{N}}$, and the decision boundary $\Gamma_{\mathcal{N}}$ of $\mathcal{N}$. The intuition behind these concepts is that x belongs to the confidence set of $\mathcal{N}$ if the output $\mathcal{N}(x)$ is one of the possible one-hot vectors. If the output is a vector where the maximum is reached in exactly one coordinate, we say that x belongs to the classified set of $\mathcal{N}$. Otherwise, the output is a vector where the maximum is reached in two or more coordinates, i.e., the instance has equal probability to belong to two or more output classes, then we say that x belongs to the decision boundary of $\mathcal{N}$. Let us observe that $T_{\mathcal{N}} \subseteq C_{\mathcal{N}}$ and $C_{\mathcal{N}} \sqcup \Gamma_{\mathcal{N}} = X$.

Definition 14. *Let $d, k > 0$ be integers. Let $D \subset \mathbb{R}^d \times \mathbb{E}^k$ be a labelled dataset and $\mathcal{N} : X \to Y$ a neural network that characterizes D. Let $x \in X$, with $\mathcal{N}(x) = (y_1, \ldots, y_k) \in Y$. If there exists $j \in [\![1, k]\!]$ such that $y_j > \max\{y_i : i \in [\![1, k]\!], i \neq j\}$, then we say that x belongs to the set $C^j_{\mathcal{N}}$ and it has label $e_j \in \mathbb{E}^k$ (with probability y_j). Moreover, we define $C_{\mathcal{N}}$ to be the union of the sets $C^j_{\mathcal{N}}$ for $j \in [\![0, k]\!]$. Besides, when $y_j = 1$, we say that x belong to the confidence set $T_{\mathcal{N}}$. Finally, we say that x belongs to the decision boundary $\Gamma_{\mathcal{N}}$ if there exists $j \in [\![1, n]\!]$ such that $y_j = \max\{y_i : i \in [\![1, k]\!], i \neq j\}$.*

The following is a key result to define a simplicial-map neural network that characterizes a given labelled dataset.

Proposition 3. *Let $d, k > 0$ be integers. Let L be the simplicial complex with only one maximal k-simplex $\sigma = \{v_0, \ldots, v_k\}$ with $v_i = e^k_i \times 0$ for $i \in [\![1, k]\!]$ and $v_0 = e^k_0 \times 1$. Let $D \subset \mathbb{R}^d \times \mathbb{E}^k$ be a labelled dataset and let $V_{\mathcal{P}}$ be the vertices of a convex polytope $\mathcal{P}$ such that $D_P \subset \mathcal{P}$. Let us assume that D_P is in general position. Let $K = \mathscr{D}(D_P \cup V_{\mathcal{P}})$. Then, the map $\varphi^{(0)} : K^{(0)} \to L^{(0)}$ defined as follows is a vertex map:*

$$\varphi^{(0)}(u) := \begin{cases} \ell \times 0 & \text{if } (u, \ell) \in D, \\ v_0 & \text{if } u \in V_{\mathcal{P}}. \end{cases}$$

Proof. L is composedof a maximal simplex. Any subset of vertices of a simplex is a simplex by definition. Then, any map between vertices of $K^{(0)}$ to $L^{(0)}$ is a vertex map. Specifically, $\varphi^{(0)}$ is a vertex map. □

By abuse of notation, we will say that a point $y \in \mathbb{R}^k$ with barycentric coordinates $(y_0, \ldots, y_k)$ has label $j \in [\![0, k]\!]$ if $y_j > \max\{y_i : i \in [\![0, k]\!], i \neq j\}$. Let us notice that an *unknown* label has been assigned to the vertex v_0 of L.

Proposition 4. *Let $\varphi^{(0)} : K^{(0)} \to L^{(0)}$ be the vertex map defined in Proposition 3. Then, the simplicial-map neural network $\mathcal{N}_\varphi : |K| \to |L|$ induced by the simplicial map φ^c characterizes D.*

Proof. By Proposition 2, the neural network $\mathcal{N}_\varphi$ satisfies that $\mathcal{N}_\varphi(x) = \varphi^c(x)$ for all $x \in |\mathscr{D}(D_P \cup V_{\mathcal{P}})|$. Besides, let us observe that the Cartesian coordinates of $\mathcal{N}_\varphi(x)$ coincide with its barycentric coordinates. Moreover, for all $x \in D_P$, $\mathcal{N}_\varphi(x) = \varphi^{(0)}(x)$ and, by definition, $\varphi^{(0)}(x) = \ell \times 0$ when $(x, \ell) \in D$. Then, we can conclude that $\mathcal{N}_\varphi$ characterizes D. □

Again, by abuse of notation, when $\mathcal{N}_\varphi$ is a simplicial-map neural network, we will denote by T_φ, C_φ, and Γ_φ, its confidence set, classified set and decision boundary, respectively.

Remark 2. *Firstly, let us observe that, with the assumptions of Proposition 3, if $x \in \mathbb{R}^d$ belongs to the decision boundary Γ_φ then $x \in |\sigma|$ for some $\sigma \in K$ satisfying that there are at least two*

vertices in σ having different labels. Moreover, $x \in |\mu|$ for some $\mu \in \mathrm{Sd}\, K$ with all its vertices in Γ_φ. Secondly, if σ is a d-simplex in $\mathrm{Sd}\, K$, then either all its vertices belong to the confidence set T_φ or $\sigma = \sigma^1 \cup \sigma^2$ with $\sigma^1, \sigma^2 \in \mathrm{Sd}\, K$ satisfying that $|\sigma^1| \subseteq \Gamma_\varphi$ and $|\sigma^2| \subseteq T_\varphi$. Finally, if σ is a d-simplex in $\mathrm{Sd}^t K$ for $t > 0$ then either all its vertices belong to the classified subset C^j_φ for some $j \in [\![0, k]\!]$, or $\sigma = \sigma^1 \cup \sigma^2$ with $\sigma^1, \sigma^2 \in \mathrm{Sd}^t K$ satisfying that $|\sigma^1| \subseteq \Gamma_\varphi$ and $\emptyset \neq |\sigma| \setminus |\sigma^1| \subseteq C^j_\varphi$.

As the following result states, when $K = \mathscr{D}(D_P \cup V_\mathcal{P})$, we can obtain a vertex map $\varphi_t^{(0)}$ from $(\mathrm{Sd}^t K)^{(0)}$ to $(\mathrm{Sd}^t L)^{(0)}$ applying the barycentric subdivision, inducing a neural network $\mathcal{N}_{\varphi_t}$ that coincides with $\mathcal{N}_\varphi$ for any integer $t > 0$. Figure 4 illustrates these concepts.

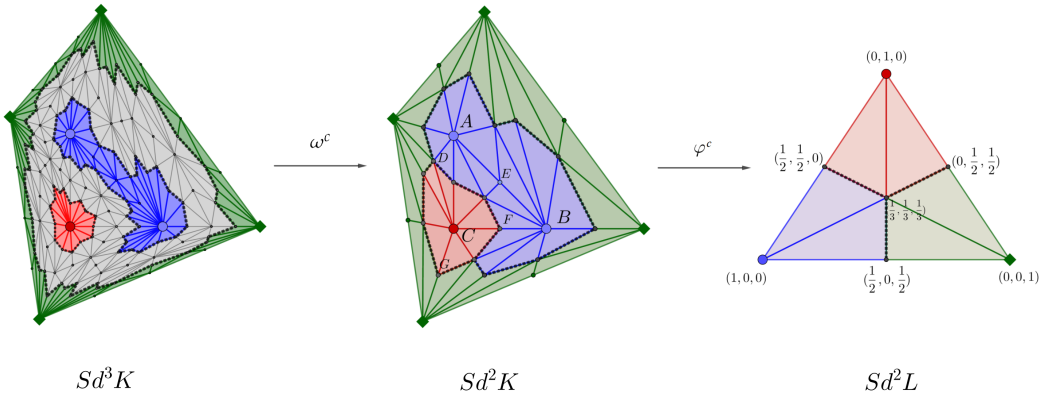


Figure 4. Let $K = \mathscr{D}(D_P \cup V_\mathcal{P})$ be the Delaunay complex of $D_P \cup V_\mathcal{P}$ where D_P is the set $\{A, B, C\}$ of red and blue points, and $V_\mathcal{P}$ are the green vertices (depicted in the center). Let L be the simplicial complex with one maximal simplex $\sigma = \{v_0 = (0,0,1), v_1 = (0,1,0), v_2 = (1,0,0)\}$ (pictured on the right). Let us consider the vertex map $\varphi^{(0)}$ that sends the blue points A, B to v_1, the red point C to v_2, and the green points (labelled as *unknown*) to v_0. Then, $\varphi^{(0)}$ gives rise to the simplicial map φ^c and the simplicial-map neural network $\mathcal{N}_\varphi$. The decision boundary of $\mathcal{N}_\varphi$ is pictured on the center as the set of points in the boundary of the red, blue or green region. For example, $\mathcal{N}_\varphi(D) = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, $\mathcal{N}_\varphi(F) = (\frac{1}{2}, \frac{1}{2}, 0)$ and $\mathcal{N}_\varphi(G) = (0, \frac{1}{2}, \frac{1}{2})$. Let us consider now the barycenter subdivision of K shown on the left and the simplicial map ω^c which relates both simplicial complexes. The decision boundary of $\mathcal{N}_{\varphi\circ\omega}$ is the gray zone on the left picture.

Lemma 1. *Let $\varphi_0^{(0)} := \varphi^{(0)}$ be the vertex map defined in Proposition 3. For an integer $t > 1$ and any $v \in (\mathrm{Sd}^t K)^{(0)}$, there exists $\mu \in \mathrm{Sd}^{t-1} K$ such that $w = \mathrm{bar}\, \mu$. Then, the map $\varphi_t^{(0)} : (\mathrm{Sd}^t K)^{(0)} \to (\mathrm{Sd}^t L)^{(0)}$ defined as:*

$$\varphi_t^{(0)}(w) := \mathrm{bar}\, \varphi_{t-1}(\mu)$$

is a vertex map inducing a neural network $\mathcal{N}_{\varphi_t}$ that coincides with $\mathcal{N}_\varphi$ for any integer $t \geq 0$.

Proof. Let $t > 0$ be an integer. Let us observe that $\varphi_0^{(0)}$ is a vertex map. By induction, let us assume that $\varphi_{t-1}^{(0)} : (\mathrm{Sd}^{t-1} K)^{(0)} \to (\mathrm{Sd}^{t-1} L)^{(0)}$ is a vertex map. Let $\sigma \in \mathrm{Sd}^t K$. By definition of barycentric subdivision, we can assume that

$$\sigma = \{w_0, \ldots, w_k\} \text{ with } w_i = \mathrm{bar}\, \mu_i, \text{ being } \mu_i \text{ a face of } \mu_j \in \mathrm{Sd}^{t-1} K \text{ for } i, j \in [\![0, k]\!] \text{ and } i < j.$$

Then,

$$\{v \in \mathrm{Sd}^t L : \exists\, w \in \sigma,\ \varphi_t^{(0)}(w) = v\} = \{\varphi_t^{(0)}(w_i) : i \in [\![0,k]\!]\} = \{\mathrm{bar}\,\varphi_{t-1}(\mu_i) : i \in [\![0,k]\!]\}.$$

Since $\varphi_{t-1}^{(0)}$ is a vertex map, then $\varphi_{t-1}(\mu_i)$ is a simplex of $\mathrm{Sd}^{t-1} L$ and $\varphi_{t-1}(\mu_i)$ is a face of $\varphi_{t-1}(\mu_j)$ for all $i, j \in [\![0,k]\!]$ with $i < j$, by definition of φ_{t-1}. Then, $\{\mathrm{bar}\,\varphi_{t-1}(\mu_i)$: $i \in [\![0,k]\!]\}$ is a simplex of $\mathrm{Sd}^t L$.

Now, let us see that $\mathcal{N}_{\varphi_t} = \mathcal{N}_{\varphi}$. By induction, let us prove that $\mathcal{N}_{\varphi_t} = \mathcal{N}_{\varphi_{t-1}}$. Let $x \in |K|$. Then, there exist a d-simplex $\mu = \{w_0, \ldots, w_d\} \in \mathrm{Sd}^t K$ and a d-simplex $\sigma = \{u_0, \ldots, u_d\} \in \mathrm{Sd}^{t-1} K$ such that $x \in |\mu| \subset |\sigma|$ and $w_i = \mathrm{bar}\{u_0, \ldots, u_i\}$ for all $i \in [\![0,d]\!]$.

Then,

$$\mathcal{N}_{\varphi_t}(x) = \varphi_t^c(x) = \sum_{i \in [\![0,d]\!]} \lambda_i \varphi_t^{(0)}(w_i)$$

being $\lambda_i \in [0,1]$ for all $i \in [\![0,d]\!]$ and $\sum_{i \in [\![0,d]\!]} \lambda_i = 1$. Therefore,

$$\mathcal{N}_{\varphi_t}(x) = \sum_{i \in [\![0,d]\!]} \lambda_i \sum_{j \in [\![0,i]\!]} \frac{1}{i+1} \varphi_t^{(0)}(u_j) = \sum_{i \in [\![0,d]\!]} \lambda_i' \varphi_t^{(0)}(u_i)$$

being $\lambda_i' = \sum_{j \in [\![i,d]\!]} \frac{\lambda_j}{j+1}$. Let us observe that

$$\sum_{i \in [\![0,d]\!]} \lambda_i' = \sum_{i \in [\![0,d]\!]} (i+1) \frac{\lambda_i}{i+1} = \sum_{i \in [\![0,d]\!]} \lambda_i = 1.$$

Now, let us observe that

$$\lambda_i' = \sum_{j \in [\![i,d]\!]} \frac{\lambda_j}{j+1} \geq 0 \quad \text{and} \quad \sum_{j \in [\![i,d]\!]} \frac{\lambda_j}{j+1} \leq \frac{1}{i+1} \sum_{j \in [\![i,d]\!]} \lambda_j \leq \frac{1}{i+1} \leq 1,$$

for all $i \in [\![0,d]\!]$. Then, for all $x \in |K|$,

$$\mathcal{N}_{\varphi_t}(x) = \sum_{i \in [\![0,d]\!]} \lambda_i' \varphi_t^{(0)}(u_i) = \mathcal{N}_{\varphi_{t-1}}(x),$$

with $\lambda_i' \in [0,1]$ for all $i \in [\![0,d]\!]$ and $\sum_{i \in [\![0,d]\!]} \lambda_i' = 1$, concluding the proof. □

Computing Simplicial-Map Neural Networks Robust to Adversarial Attacks

In this subsection, the main result of the paper is provided. It states that we can always compute a neural network characterizing a given labelled dataset, and being robust to adversarial attacks of a given size. Firstly, let us define the concepts of adversarial example and robustness of neural networks against adversarial attacks. Some interesting references on these concepts are [21,22].

Definition 15. *Let $d, k > 0$ be integers. Let $D \subset \mathbb{R}^d \times \mathbb{E}^k$ be a labelled dataset and $\mathcal{N}$ a neural network that characterizes D. Let $B(r) = \{\alpha \in \mathbb{R}^d : ||\alpha|| \leq r\}$ being $||\cdot||$ a norm on $\mathbb{R}^d$. Let us suppose that $x \in \mathbb{R}^d$ has label ℓ. Then, an adversarial example of size r is defined as $x' = x + \alpha$ with $\alpha \in B(r)$ such that x' has label ℓ' with $\ell' \neq \ell$. A neural network is called robust to adversarial attacks of size r if no labelled point $x \in \mathbb{R}^d$ has an adversarial example of size r.*

Proposition 5. *With the assumptions of Proposition 4, we have that $\mathcal{N}_{\varphi}$ is not robust to adversarial attacks of size r for $0 < r < d(T_{\varphi}, \Gamma_{\varphi})$.*

Proof. By Remark 2, consider $\sigma \in K$ such that there exist v and w being two vertices of σ with different labels. Let $z = \mathrm{bar}\{v, w\}$. Then z is in the decision boundary of $|K|$ and

$\{v,z\}$, $\{z,w\}$ are edges of Sd K. Let $x=(1-a)z+av$ where $a=\frac{r}{2d(z,v)}$. Then x has the same label as v and $d(x,z)=\frac{r}{2}$. Let $x'=(1-a')z+a'w$ where $a'=\frac{r}{2d(z,w)}$. Then x' has the same label as w and $d(z,x')=\frac{r}{2}$. Then, $d(x,x')=d(x,z)+d(z,x')=r$ concluding that x' is an adversarial example of size r and $0<r<d(T_\varphi,\Gamma_\varphi)$. □

Example 1. *Let $d=k=2$. Let us consider the labelled dataset $D=\{(A=(4,8),(0,1,0)),(B=(8,4),(1,0,0))\}$ and the convex polytope $\mathcal{P}$ with vertices $\{C=(5,10),D=(0,0),E=(15,0)\}$. Then, $K=\mathscr{D}(D_P\cup V_\mathcal{P})$ is composed by five maximal 2-simplices and L by just one maximal 2-simplex. This way,* Sd K *and* Sd L *are composed by the maximal 2-simplices showed in Figure 5. Let $x=(6-a,6+a)\in|K|$, with $a\in(0,2]$, be in the geometric realization of the segment with endpoints $\{(4,8),(8,4)\}$. Then, $\varphi^c(x)=(c,d,0)$ with $0\leq c<\frac{1}{2}$ and $\frac{1}{2}<d\leq 1$. Therefore, x is classified as $(0,1,0)$ with probability d. Take $z=(6,6)$. Then, z belongs to the decision boundary since $\varphi^c(z)=(\frac{1}{2},\frac{1}{2},0)$. Take $x'=(6+a,6-a)$. Then, $\varphi^c(x)=(c',d',0)$ with $\frac{1}{2}<c'\leq 1$ and $0\leq d'<\frac{1}{2}$. Therefore, x' is classified as $(1,0,0)$ with probability d'.*

Since $d(x,z)=a\sqrt{2}=d(x',z)$ and $a\in(0,2]$, then $\mathcal{N}_\varphi$ is not robust to adversarial attacks of any size r with $0<r\leq 4\sqrt{2}$. See Figure 5.

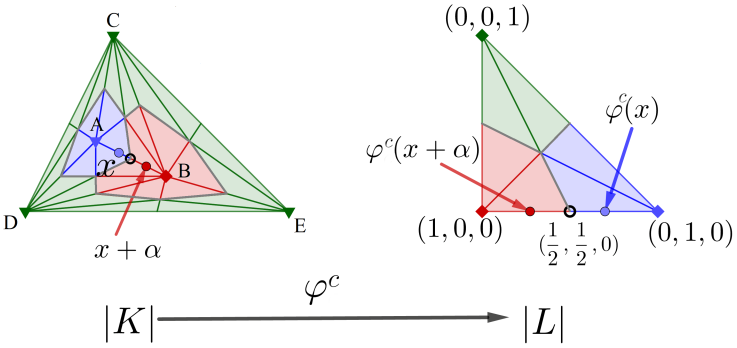


Figure 5. An adversarial example x for the simplicial-map neural network $\mathcal{N}_\varphi:|K|\to|L|$.

Let us now introduce the main result of this paper stating that there exists a two-hidden-layer neural network characterizing a given labelled dataset and being robust to adversarial attacks of size $r>0$ for r being *small enough*. In order to define such a neural network robust to adversarial examples, we will construct a continuous function from $|K|$ to $|K|$ with the idea of later applying the Simplicial Approximation Theorem and the composition of simplicial maps to obtain a simplicial map from $|K|$ to $|L|$ that will give rise to a neural network robust to adversarial attacks of a given size $r>0$. Let us observe that, to be able to compute such a robust neural network, the size r should be smaller than the distance between the decision boundary and the confidence set.

Theorem 3. *Let $n,d,k>0$ be integers. Let $D=\{(p_j,\ell_j)\colon j\in[\![1,n]\!],p_j\in\mathbb{R}^d,\ell_j\in\mathbb{E}^k\}$ be a labelled dataset. Then, there exists a two-hidden-layer neural network $\mathcal{N}$ characterizing D and robust to adversarial attacks of size $r>0$, for r being small enough.*

Proof. Let us consider a convex polytope $\mathcal{P}$ such that the points of D_P are inside $\mathcal{P}$. Then, we can compute the Delaunay complex $\mathscr{D}(D_P\cup V_\mathcal{P})$ that will be denoted simply by K (see Figure 4), and a simplicial complex L composed of just one maximal k-simplex. As claimed in Proposition 4, a simplicial map φ^c can be defined between $|K|$ and $|L|$ giving rise to a neural network $\mathcal{N}_\varphi$ that characterizes D (see Proposition 2). However, $\mathcal{N}_\varphi$ is not robust to adversarial attacks (see Proposition 5). Our goal is to define a new simplicial map such that its associated simplicial-map neural network is robust to adversarial attacks. To reach that aim, we need r to be small enough, that is, $0<r<d(T_\varphi,\Gamma_\varphi)$, where $d(T_\varphi,\Gamma_\varphi)=\min\{d(p,q):p\in T_\varphi,q\in\Gamma_\varphi\}$, so adversarial attacks will be placed between the confidence set T_φ and the decision boundary Γ_φ. Then, a continuous function $g:|\text{Sd}\,K|\to|\text{Sd}\,K|$

will be defined depending on such r, to later apply the Simplicial Approximation Theorem Extension (Proposition 1), obtaining a simplicial approximation $\omega^c : \mathrm{Sd}^s K \to \mathrm{Sd}^t K$ of g as close to g as desired. Then, $\varphi_t^c \circ \omega^c : |K| \to |L|$ will be a simplicial map giving rise to a simplicial-map neural network $\mathcal{N}_{\varphi_t \circ \omega}$ robust to adversarial attacks of size r.

Let us define now the continuous function $g : |\mathrm{Sd}\, K| \to |\mathrm{Sd}\, K|$. Let $\sigma = \{u_0, \ldots, u_d\}$ be a d-simplex of $\mathrm{Sd}\, K$. Let us observe that, by Remark 2, the vertices of σ satisfy the following property:

- All the vertices of σ are in T_φ. Then, $|\sigma| \subseteq T_\varphi$.
- Otherwise, $\sigma = \sigma^1 \cup \sigma^2$ being $\emptyset \neq |\sigma^1| \subseteq \Gamma_\varphi$ and $\emptyset \neq |\sigma^2| \subseteq T_\varphi$.

In the latter case, let us define the continuous function $g : |\sigma| \to |\sigma|$ as follows. Without loss of generality, let us assume that $\sigma^1 = \{u_0, \ldots, u_h\}$ with $h \in [\![0, d]\!]$.

Let us compute the set of points of $|\sigma|$ at distance less than r to $|\sigma^1|$ and let us send, by g, such points to points in $|\sigma^1|$.

Let x be a point of $|\sigma|$ with barycentric coordinates $(x_0, \ldots, x_d)$ with respect to σ.

Let $\lambda = \sum_{i \in [\![0,h]\!]} x_i$.

Let $z^1 \in \mathbb{R}^d$ be the projection of x in $|\sigma^1|$ whose barycentric coordinates with respect to σ are:

$$(z_0, \ldots, z_h, 0, \ldots, 0), \text{ where } z_i = \tfrac{x_i}{\lambda} \text{ for } i \in [\![0, h]\!].$$

Let z^2 be the point in $|\sigma^2|$ with barycentric coordinates $(0, \ldots, 0, z_{h+1}, \ldots, z_d)$ with respect to σ, aligned with x and z^1. Then,

$$\tfrac{x_i - z_i}{x_i} = \tfrac{x_j}{x_j - z_j} \text{ for } i \in [\![0, h]\!] \text{ and } j \in [\![h+1, d]\!].$$

So, $z_j = \frac{x_j}{1-\lambda}$ for $j \in [\![h+1, d]\!]$.

Now, $x = (1-a)z^1 + az^2$ for $a \in [0, 1]$.

Then, $d(x, \sigma^1) \leq d(x, z^1) = a \cdot d(z^1, z^2)$.

Let $\varepsilon = \frac{r}{d(z^1, z^2)}$. Then, $d(x, \sigma^1) \leq r$ if $a \leq \varepsilon$. Then,

$$g(x) := \begin{cases} z^1 & \text{if } a \in [0, \varepsilon], \\ (1-a')z^1 + a'z^2 & \text{if } a \in [\varepsilon, 1] \text{ with } a' = \frac{a-\varepsilon}{1-\varepsilon}. \end{cases}$$

Let us observe that $a' \in [0, 1]$ and for $a = \varepsilon$, we have that $a' = 0$ so $(1-a')z^1 + a'z^2 = z^1$.

Besides, for $a = 1$, we have that $a' = 1$ so $(1-a')z^1 + a'z^2 = z^2$.

Let us prove that g is continuous at any point $x \in |K|$.

Let us observe that, by construction, g is continuous in the interior of $|\sigma|$ of every d-simplex $\sigma \in \mathrm{Sd}\, K$.

Let x be a point in $|\sigma \cap \mu|$ for some d-simplices $\sigma, \mu \in \mathrm{Sd}\, K$.

Let $\sigma = \sigma^1 \cup \sigma^2$ and $\mu = \mu^1 \cup \mu^2$ with $|\sigma^1|, |\mu^1| \subseteq \Gamma_\varphi$ and $|\sigma^2|, |\mu^2| \subseteq T_\varphi$. Let $\gamma \in \mathrm{Sd}\, K$ be the simplex with lower dimension such that $x \in |\gamma|$. Then, by definition of simplicial complex, $\gamma \subseteq \sigma \cap \mu$ and the barycentric coordinates of x with respect to σ and μ coincide.

By Remark 2, we have to consider two cases:

(1) All the vertices of γ belong to T_φ. Then $\gamma \subseteq \sigma^2 \cap \mu^2$ and $g(x) = x$.
(2) $\gamma = \gamma^1 \cup \gamma^2$ with $|\gamma^1| \subseteq \Gamma_\varphi$ and $|\gamma^2| \subseteq T_\varphi$. Then $\gamma^1 \subseteq \sigma^1 \cap \mu^1$ and $\gamma^2 \subseteq \sigma^2 \cap \mu^2$ so the definition of $g(x)$ with respect to σ and μ coincides.

Now, by Proposition 1, given $r_1 > 0$, there exist $s, t > 0$ and a simplicial map $\omega^c : |\mathrm{Sd}^s K| \to |\mathrm{Sd}^t K|$ such that $||g - \omega^c|| < r_1$. By Lemma 1, $\varphi_t^{(0)} : (\mathrm{Sd}^t K)^{(0)} \to (\mathrm{Sd}^t L)^{(0)}$ is a vertex map. Since the composition of simplicial maps is a simplicial map, then $\varphi_t^c \circ \omega^c : |\mathrm{Sd}^s K| \to |\mathrm{Sd}^t L|$ is a simplicial map, concluding that $\mathcal{N}_{\varphi_t \circ \omega}$ is a simplicial-map neural network.

Let us prove now that $\mathcal{N}_{\varphi_t \circ \omega}$ is robust to adversarial attacks of size r. First of all, the following properties holds:

(1) If $x \in C^j_{\varphi_t \circ \omega}$ then $\omega^c(x) \in C^j_{\varphi_t}$, being $j \in [\![0,k]\!]$.

(2) Let $v \in (\mathrm{Sd}^t K)^{(0)}$ and $z \in |\operatorname{st} v|$. If $v \in C^j_{\varphi_t}$ then $z \in C^j_{\varphi_t}$, being $j \in [\![0,k]\!]$.
Since $z \in |\operatorname{st} v|$ then $z \in |\sigma|$ for a d-simplex $\sigma \in \mathrm{Sd}^t K$ with $v \in \sigma$. Then, by Remark 2, $\sigma = \sigma^1 \cup \sigma^2$, with $\sigma^1 \in \Gamma_{\varphi_t}$ and $\sigma^2 \in C^j_{\varphi_t}$. Besides, since $v \in \sigma^2$ then $z \notin |\sigma^1|$, therefore $z \in |\sigma| \setminus |\sigma^1 \subseteq C^j_{\varphi_t}$.

(3) If $x \in C^j_{\varphi_t \circ \omega}$ then $g(x) \in C^j_{\varphi_t}$, being $j \in [\![0,k]\!]$.
If $x \in C^j_{\varphi_t \circ \omega}$ then there exists $v \in C^j_{\varphi_t \circ \omega}$ such that $x \in |\operatorname{st} v|$. Then, $\omega(v) \in C^j_{\varphi_t}$ by (0). Now, since $g(|\operatorname{st} v|) \subseteq |\operatorname{st} \omega(v)|$, then $g(x) \in C^j_{\varphi_t}$ by (1).

(4) Let $x \in |\mathrm{Sd}^s K|$. If $g(x) \in \Gamma_{\varphi_t}$ then $x \in \Gamma_{\varphi_t \circ \omega}$.
By contradiction, let us assume that $g(x) \in \Gamma_{\varphi_t}$ and $x \in C^j_{\varphi_t \circ \omega}$ for some $j \in [\![0,k]\!]$. Then, $g(x) \in C^j_{\varphi_t}$ by (2), leading to a contradiction.

(5) Let $x \in |\mathrm{Sd}^s K|$. If $g(x) \in C^j_{\varphi_t}$ with probability y_j and $x \in C^{j'}_{\varphi_t \circ \omega}$ then $j = j'$ and $|y_j - y'_j| < r_1$. This last statement is a consequence of (2) and that $||g - \omega^c|| < r_1$.

Now, let $x \in C^j_{\varphi_t \circ \omega}$ being $j \in [\![0,k]\!]$. Let $\alpha \in \mathbb{R}^d$ with $||\alpha|| < r$ and let $x' := x + \alpha$. Let us prove that $x' \in \Gamma_{\varphi_t \circ \omega}$ or $x' \in C^j_{\varphi_t \circ \omega}$.

On one hand, if $g(x') \in \Gamma_{\varphi_t}$ then $x' \in \Gamma_{\varphi_t \circ \omega}$ by (4). On the other hand, if $g(x') \in C^j_{\varphi_t}$ then $x' \in C^j_{\varphi_t \circ \omega}$ or $x' \in \Gamma_{\varphi_t \circ \omega}$ by (3), concluding the proof. □

Example 2. *Let us consider a labelled dataset $D = \{(p,1)\}$ with $p \in \mathbb{R}$ composed of just one point. Let $\mathcal{P}$ be a segment with endpoints p_1 and p_2 in $\mathbb{R}$ such that $p_1 < p < p_2$. Let $r \in \mathbb{R}$ such that $0 < r < \min\{|p_1 - p|, |p_2 - p|\}$. Let K be the Delaunay complex of $\{p, p_1, p_2\}$ that consists of just the two maximal simplices $\{p, p_1\}$ and $\{p, p_2\}$. Let L be a simplicial complex composed by a maximal 1-simplex with endpoints $v_0 = (0,1)$ and $v_1 = (1,0)$. Then, a simplicial map φ^c can be defined as in Proposition 3 together with a neural network $\mathcal{N}_\varphi$ as in Proposition 4. However, $\mathcal{N}_\varphi$ is not robust to attacks of size r as it has been proved in Proposition 5. Then, following the proof of Theorem 3, in Figure 6, we have computed barycentric subdivisions on K until we approximate g by the simplicial map $\omega^c : |\mathrm{Sd}^3 K| \to |\mathrm{Sd}^2 K|$. Finally, the neural network induced by the composition $\varphi_2^c \circ \omega^c$ is robust to adversarial attacks of size r.*

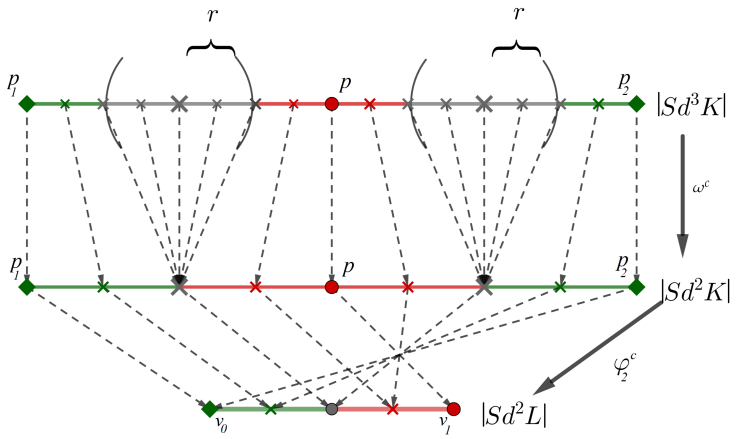


Figure 6. Three simplicial complexes with simplicial maps ω^c and φ_2^c between them are shown illustrating a neural network $\mathcal{N}_{\varphi_2 \circ \omega}$ robust to adversarial attacks of size r.

5. Conclusions and Future Work

Neural networks are one of the most promising tools in artificial intelligence and, currently, with the big success on real-world problem of Deep Learning architectures, it has become one of the most widely used. From a mathematical point of view, neural network can be seen as the composition of a big amount of simple functions, mainly from linear

algebra, and the so-called activation functions. Since the efficiency of such neural networks depends of the choice of an appropriate set of parameters, most of the efforts in the study of such networks has been focused on optimization techniques. After a first wave of research based on these optimization techniques, many researchers are considering the study of neural networks by using different mathematical techniques as analysis, geometry or, as in this paper, algebraic topology.

Specifically, in this paper, we have presented a family of neural networks, called simplicial-map neural networks, that are robust to adversarial examples. The main contribution of the paper is a constructive proof that shows how to define a neural network robust to adversarial attacks of a given size. This result is proven thanks to the connection of neural networks with concepts from Algebraic Topology. By endowing the set of instances of a classification problem with the structure of simplicial complex and considering the set of one-hot labels as a simplex, provides a new point of view that allows to find the exact values of the weights of the associated network without any kind of training or optimization process.

Finally, we plan to provide an implementation of our methods that takes into account the efficiency issues that arise in creating neural networks following our approach. We believe that this point of view opens a new bridge between Neural Network and Algebraic Topology which can lead to a fruitful flow of concepts, problems and solutions in both directions.

Author Contributions: Conceptualization, R.G.-D., M.A.G.-N., J.H. and E.P.-H.; methodology, R.G.-D., M.A.G.-N., J.H. and E.P.-H.; formal analysis, R.G.-D., M.A.G.-N., J.H. and E.P.-H.; investigation, R.G.-D., M.A.G.-N., J.H. and E.P.-H.; writing—original draft preparation, R.G.-D., M.A.G.-N., J.H. and E.P.-H.; writing—review and editing, R.G.-D., M.A.G.-N., J.H. and E.P.-H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by MICINN, FEDER/UE under grant PID2019-107339GB-100.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing properties of neural networks. *arXiv* **2013**, arXiv:cs.CV/1312.6199.
2. Fezza, S.A.; Bakhti, Y.; Hamidouche, W.; Déforges, O. Perceptual Evaluation of Adversarial Attacks for CNN-based Image Classification. In Proceedings of the Eleventh International Conference on Quality of Multimedia Experience (QoMEX), Berlin, Germany, 5–7 June 2019; pp. 1–6. [CrossRef]
3. Garg, S.; Ramakrishnan, G. BAE: BERT-based Adversarial Examples for Text Classification. *arXiv* **2020**, arXiv:cs.CL/2004.01970.
4. Karim, F.; Majumdar, S.; Darabi, H. Adversarial Attacks on Time Series. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, 1. [CrossRef] [PubMed]
5. Christakopoulou, K.; Banerjee, A. Adversarial Attacks on an Oblivious Recommender. In Proceedings of the 13th ACM Conference on Recommender Systems, Association for Computing Machinery, Copenhagen, Denmark, 20 September 2019; pp. 322–330. [CrossRef]
6. Xu, H.; Ma, Y.; Liu, H.; Deb, D.; Liu, H.; Tang, J.; Jain, A.K. Adversarial Attacks and Defenses in Images, Graphs and Text: A Review. *Int. J. Autom. Comput.* **2020**, *17*, 151–178. [CrossRef]
7. Yan, Z.; Guo, Y.; Zhang, C. Adversarial Margin Maximization Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, 1. [CrossRef] [PubMed]
8. Cortes, C.; Vapnik, V. Support Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
9. Tang, Y. Deep Learning using Linear Support Vector Machines. *arXiv* **2013**, arXiv:cs.LG/1306.0239.
10. Sun, S.; Chen, W.; Wang, L.; Liu, X.; Liu, T. On the Depth of Deep Neural Networks: A Theoretical View. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; Schuurmans, D., Wellman, M.P., Eds.; AAAI Press: Palo Alto, CA, USA, 2016; pp. 2066–2072.
11. Wang, X.; Zhang, S.; Lei, Z.; Liu, S.; Guo, X.; Li, S.Z. Ensemble Soft-Margin Softmax Loss for Image Classification. In Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI'18, Stockholm, Sweden, 13–19 July 2018; AAAI Press: Palo Alto, CA, USA, 2018; pp. 992–998.

12. Paluzo-Hidalgo, E.; Gonzalez-Diaz, R.; Gutiérrez-Naranjo, M.A. Two-hidden-layer feed-forward networks are universal approximators: A constructive approach. *Neural Netw.* **2020**, *131*, 29–36. [CrossRef] [PubMed]
13. Ismailov, V.E. On the approximation by neural networks with bounded number of neurons in hidden layers. *J. Math. Anal. Appl.* **2014**, *417*, 963–969. [CrossRef]
14. Guliyev, N.J.; Ismailov, V.E. Approximation capability of two hidden layer feedforward neural networks with fixed weights. *Neurocomputing* **2018**, *316*, 262–269. [CrossRef]
15. Ebli, S.; Defferrard, M.; Spreemann, G. Simplicial Neural Networks. *arXiv* **2020**, arXiv:cs.LG/2010.03633.
16. Spanier, E.H. *Algebraic Topology*; Springer: New York, NY, USA, 1995.
17. Boissonnat, J.D.; Chazal, F.; Yvinec, M. *Geometric and Topological Inference*; Cambridge Texts in Applied Mathematics; Cambridge University Press: Cambridge, UK, 2018. [CrossRef]
18. Okabe, A.; Boots, B.; Sugihara, K.; Chiu, S.N.; Kendall, D.G. Definitions and Basic Properties of Voronoi Diagrams. In *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, 2nd ed.; John Wiley & Sons: Chichester, UK, 2000; pp. 43–112. [CrossRef]
19. Hornik, K. Approximation Capabilities of Multilayer Feedforward Networks. *Neural Netw.* **1991**, *4*, 251–257. [CrossRef]
20. Edelsbrunner, H.; Harer, J. *Computational Topology—An Introduction*; American Mathematical Society: Providence, RI, USA, 2010; pp. 1–241.
21. Lecuyer, M.; Atlidakis, V.; Geambasu, R.; Hsu, D.; Jana, S. Certified Robustness to Adversarial Examples with Differential Privacy. In Proceedings of the IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 18–19 May 2019; pp. 656–672. [CrossRef]
22. Yuan, X.; He, P.; Zhu, Q.; Li, X. Adversarial Examples: Attacks and Defenses for Deep Learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 2805–2824. [CrossRef] [PubMed]