A tropical approach to neural networks

Grégoire Fournier

September 2019

Applied Mathematics Master thesis
Department of Mathematics
Supervised by A. Broustet

Abstract

This thesis takes an algebraic approach to the study of neural networks with Relu activations. Through the characterization of tropical hypersurfaces associated to networks, it may be possible to better understand performances and the training efficiency on different architectures. It gives indications on how pick an architectures regarding the expressivity or the types of hypersurfaces and frontiers of decision that the networks aims at modeling.

1 Introduction

Tropical geometry is a recent subfield based on a **max plus algebra**. Recently its link with neural networks has been formalized by Zhang et al. [16]. This thesis aims to highlight the links between neural networks and tropical geometry. It offers a first implementation of these ideas and presents some results from this rapprochement.

First this thesis shows the link between neural networks and tropical algebra and its consequences on the complexity of decision boundaries. Then is presented some algebraic theory to better understand the geometric varieties involved and on their computer representation. Finally, some results from the algebraic approach are presented as well as an example of a practical application to a classification problem.

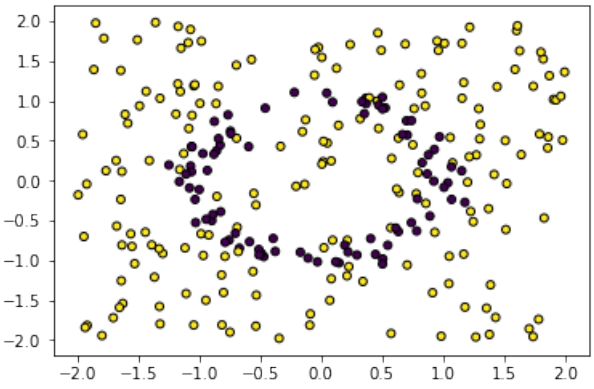

Figure 1: Example of a classification problem, a 2d donut

2 Tropical Algebra and neural networks

2.1 Introduction to neural networks

In this part we want to show that under certain assumptions, there is an equivalence between a neural network and a tropical rational function. We will only consider fully connected neural networks. A neural network is the successive composition of linear functions and nonlinear functions called activations. These activation functions are fixed and the training of the network consists in optimizing an objective function by modifying the linear functions.
A network nu of depth L is therefore written $\mathbb{R}^{n_1} \to \mathbb{R}^{n_L}$:

$$\nu^{(L)} = \sigma^{(L)} \circ \rho^{(L)} \circ \sigma^{(L-1)} \circ ... \circ \sigma^{(1)} \circ \rho^{(1)}$$

Where ρ are the linear functions and σ the activations.

Each couple $(\rho^{(l)}, \sigma^{(l)})$ is called a layer. $\rho^{(l)} : \mathbb{R}^{n_{l-1}} \to \mathbb{R}^{n_l}$ is defined by :

$$\rho^{(l)}(\boldsymbol{x}) := A^{(l)}\boldsymbol{x} + \boldsymbol{b}^{(l)}$$

Where $A^{(l)} \in \mathbb{R}^{n_l \times n_{l-1}}$ are called the weight matrices, $\boldsymbol{b}^{(l)} \in \mathbb{R}^{n_l}$ the biases.

One of the most popular activations is *Relu*, *rectified linear unit* which equals to $max(0, .)$ fig. 2.
he first transformation can be written by applying the max component by component $\sigma(\rho(\boldsymbol{x})) = max(A\boldsymbol{x} + b, 0)$.
Subsequently σ will be the function $max(0, .)$ and can be applied in any dimension by applying the transformation $max(0, .)$ component by component.

2.2 The tropical semi ring

Tropical algebra is the study of the semi-ring $(\mathbb{R} \cup \{-\infty\}, \oplus, \odot)$.

In the rest of the paper we will use $\mathbb{T} := \mathbb{R} \cup \{-\infty\}$

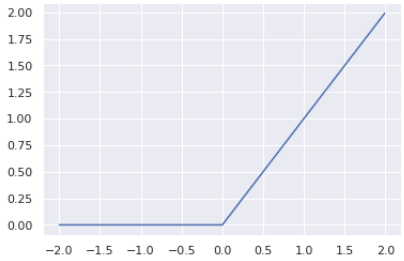

Figure 2: *Relu* Activation

We define two associative operations on $\mathbb{T}$: $x \oplus y := max(x, y)$ and $x \odot y := x + y$

Tropical division is also defined $\oslash : x, y \to x - y$

This space extends by taking addition, multiplication and multiplication by a scalar in larger dimension with a component by component approach:

$$\mathbb{T}^d \times \mathbb{T}^d \to \mathbb{T}^d$$

$$\boldsymbol{x} \odot \boldsymbol{y} = x_i \odot y_i$$

Tropical power is written $x^{\odot a} := x \odot ... \odot x = a \times x$

We will simply write x^a for tropical power when there is no ambiguity about the tropical nature of the equation. Only integer powers will be considered.

We note that $-\ infty$ and resp. 0 are the neutrals for *oplus* and resp. *odot*. And that the structure defined is a semi-ring by the associativity, commutativity and distributivity of these operators. The inverse for the addition is not defined.

2.3 Tropical polynomials and tropical rational functions

n dimension d, for $\boldsymbol{a}\ in\mathbb{N}^d$, a tropical monomial, denoted $\boldsymbol{x^a}$, is an expression of the form:

$$\boldsymbol{x^a} = c \odot {x_1}^{a_1} \odot ... \odot {x_d}^{a_d}$$

$$\mathbb{T}^d \to \mathbb{T}$$

A tropical polynomial f in $athbbR^d$ is the tropical sum, denoted $\sum$ of tropical monomials:

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{a} \in S} \boldsymbol{x^a} = \sum_{\boldsymbol{a} \in S} c_{\boldsymbol{a}} \odot x_1^{a_1} \odot ... \odot x_d^{a_d}$$

for S a finite subset of $\mathbb{N}^d$, c reals, for all $\boldsymbol{x}$ of $\mathbb{R}^d$.

A tropical rational function is the difference of two tropical polynomials:

$$F(\boldsymbol{x}) = P_1(\boldsymbol{x}) - P_2(\boldsymbol{x})$$

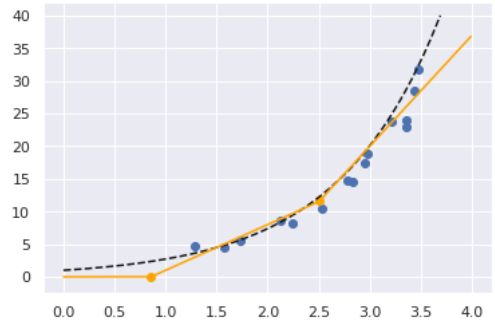

Figure 3: Regression problem of the exponential function $f(x) = 0 \oplus (-6 \odot x^7) \oplus (-31 \odot x^{17})$

In the fig. 3 we show how we can use the space of tropical polynomials to approximate the exponential function.

These expressions and the subdivisions of the space they entail have been present for some years in the field of machine learning.

The use of 'max' activations is widespread in the machine learning community. Indeed, as it is shown later, the choice of the tropical functions as basis of representation allows us to give an equivalence between a tropical rational function and the Relu activation in the neural networks.

2.4 Tropical Rational Functions and Neural Networks

We set 3 conditions on the elements of the neural networks to obtain an equivalence with the tropical functions :

(a) weight matrices are integer valued

(b) biases are real valued

(c) the activations are of type *Relu*

Proposition 2.1. *Tropical characterization of layer-to-layer transformations in a neural network. A neural network fulfilling the conditions (a) - (c) can be written as a tropical rational function.*

Proof. Writing $A^{(1)} \in \mathbb{Z}^{n_2 \times n_1}$ and $\boldsymbol{b^{(1)}} \in \mathbb{R}^{n_2}$ the matrices of weights associated with the first layer, we break down $A^{(1)}$ in 2 matrices with positive coefficients : $A^{(1)} = A^{(1)}_+ - A^{(1)}_-$

The output of the first layer is :

$$\nu^{(1)} = \sigma \circ \rho^{(1)}(\boldsymbol{x}) = max(A\boldsymbol{x} + \boldsymbol{b}, \boldsymbol{0}) = max(A^{(1)}_+\boldsymbol{x} - A^{(1)}_-\boldsymbol{x} + \boldsymbol{b}, \boldsymbol{0})$$

$$\nu^{(1)} = \sigma \circ \rho^{(1)}(\boldsymbol{x}) = max(A^{(1)}_+\boldsymbol{x} + \boldsymbol{b}, A^{(1)}_-\boldsymbol{x}) - A^{(1)}_-\boldsymbol{x}$$

We then set the first member as $F^{(1)}$ and the second as $G^{(1)}$, we notice that $F^{(1)}$ and $G^{(1)}$ are tropical polynomials.

$$\nu^{(1)} = \sigma \circ \rho^{(1)}(\boldsymbol{x}) = F^{(1)}(\boldsymbol{x}) - G^{(1)}(\boldsymbol{x})$$

$\nu^{(1)}$ is therefore a tropical rational function as difference of two tropical polynomials.
If at step l the neural network can be written as a tropical rational function:

$$\nu^{(l)} = \sigma \circ \rho^{(l)}(\boldsymbol{x}) = F^{(l)}(\boldsymbol{x}) - G^{(l)}(\boldsymbol{x})$$

Writing $A^{(l+1)} \in \mathbb{Z}^{n_{l+1} \times n_l}$ and $\boldsymbol{b^{(l+1)}} \in \mathbb{R}^{n_{l+1}}$ the matrices of weight associated with the $l+1$th layer,

$$\rho^{(l+1)} \circ \nu^{(l)}(\boldsymbol{x}) = H^{(l+1)}(\boldsymbol{x}) - G^{(l+1)}(\boldsymbol{x}))$$

$$\nu^{(l+1)} = \sigma \circ \rho^{(l+1)} \circ \nu^{(l)}(\boldsymbol{x}) = F^{(l+1)}(\boldsymbol{x}) - G^{(l+1)}(\boldsymbol{x}))$$

with :

$$H^{(l+1)}(\boldsymbol{x}) = A_+^{(l+1)} F^{(l)}(\boldsymbol{x}) + A_-^{(l+1)} G^{(l)}(\boldsymbol{x}) + \boldsymbol{b^{(l+1)}}$$
$$G^{(l+1)}(\boldsymbol{x}) = A_+^{(l+1)} G^{(l)}(\boldsymbol{x}) + A_-^{(l+1)} F^{(l)}(\boldsymbol{x})$$
$$F^{(l+1)}(\boldsymbol{x}) = max(H^{(l+1)}(\boldsymbol{x}), G^{(l+1)}(\boldsymbol{x}))$$

And we get the following tropical equations for coordinates $f_i^{(l)}, g_i^{(l)}$ and $h_i^{(l)}$ of these matrices:

$$h_i^{(l+1)} = \bigodot_{j=1}^{n_{l+1}} f_j^{(l) a_{i,j}^+} \odot \bigodot_{j=1}^{n_{l+1}} g_j^{(l) a_{i,j}^-} \odot b_i$$

$$g_i^{(l+1)} = \bigodot_{j=1}^{n_{l+1}} f_j^{(l) a_{i,j}^-} \odot \bigodot_{j=1}^{n_{l+1}} g_j^{(l) a_{i,j}^+}$$

$$f_i^{(l+1)} = h_i^{(l+1)} \oplus g_i^{(l+1)}$$

□

The next theorem follows from the previous proposition:

Theorem 2.2. *Equivalence between neural networks and tropical rational function*

(a) Given ν a neural network : $\mathbb{R}^d \to \mathbb{R}$, ν is a tropical rational function if and only if ν verifies (a) - (c)

(b) Given a tropical rational function $\nu = f \oslash g$, ν can be represented by a L layers network with

$$L \leq max(\lceil log_2(r_f) \rceil, \lceil log_2(r_g) \rceil) + 2$$

writing r_f and r_g the number of monomials in f and g

A layer in a neural network corresponds to a couple (ρ, σ).

Proof. (a) The converse comes from the previous proposition. The implication follows the same proof by construction as (b).

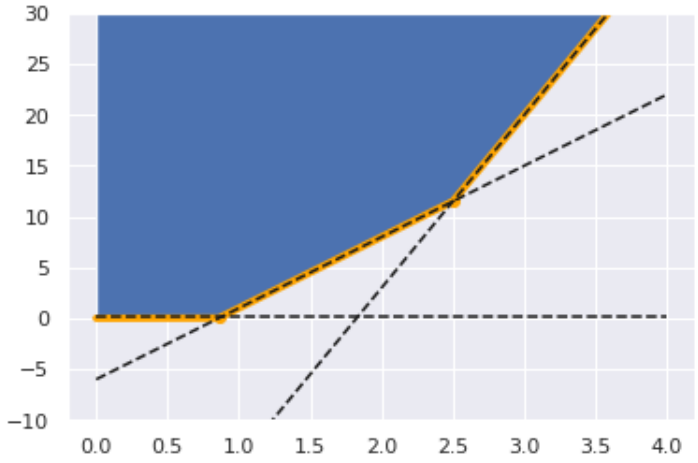

Figure 4: Dome of $f(x) = 0 \oplus x^7 \odot -6 \oplus x^{17} \odot -31$

(b) It is sufficient to build a neural network with each layer : f and g from the previous decomposition. First is initialized a neural network for a tropical monomial, then a tropical polynomial and finally a tropical rational function:

A tropical monomial can be written $\rho(\boldsymbol{x}) = \boldsymbol{b}\boldsymbol{x}^{\boldsymbol{a}}$

If p and q two tropical polynomials are written with nets of l_p and l_q layers :

$$p = \rho_p^{l_p} \circ \sigma \circ ... \circ \sigma \circ \rho_p^1 \ \ et \ \ q = \rho_q^{l_q} \circ \sigma \circ ... \circ \sigma \circ \rho_q^1$$

We can write $p \oplus q$ with $max(l_p, l_q) + 1$ layers :

$$p \oplus q = \sigma \circ (p - q) + \sigma \circ q - \sigma \circ (-q) = max(p, q)$$

By induction we conclude that p can be written with $\lceil log_2(l_p) \rceil + 1$ layers.

We can also write $p \oslash q$ with $max(l_p, l_q) + 1$ layers :

$$p \oslash q = \sigma \circ p - \sigma \circ (-p) + \sigma \circ (-q) - \sigma \circ (-q) = p - q$$

So if f and g are two tropical polynomials composed of r_f and r_g monomials respectively, then $f \oslash g$ in $max(\lceil log_2(r_f) \rceil, \lceil log_2(r_g) \rceil) + 2$ layers.

□

We associate with each polynomial f a dome which consists of all the points under the tropical curve:

$$\mathcal{D}(f) := \left\{ (\boldsymbol{p}, s) \in \mathbb{R}^{d+1} | \boldsymbol{p} \in \mathbb{R}^d, s \in \mathbb{R}, s \geq f(\boldsymbol{p}) \right\}$$

We will see that the dome of a tropical polynomial is related to its hypersurface.

Proposition 2.3. *$\mathcal{D}(f)$ is a convex polyhedron of dimension $d + 1$.*

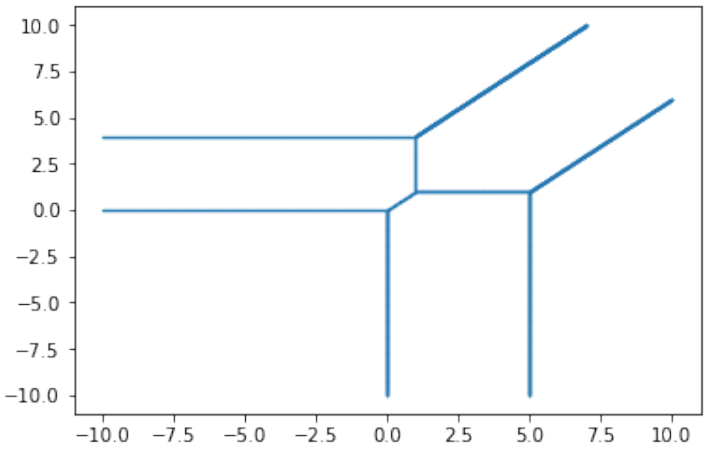

Figure 5: Tropical hypersurface of $f(x, y) = 4 + x + y \oplus 1 + 2y \oplus 2x \oplus 5 \oplus 5 + x \oplus 5 + y$

Proof. $\mathcal{D}(f)$ is the intersection of a finite number of convex spaces of $\mathbb{R}^{d+1}$ therefore is a convex polyhedron.
Let p be a point s such that $s > f(\boldsymbol{p})$, by definition of $\mathcal{D}(f)$ exists a $\varepsilon > 0$ such that $(\boldsymbol{p}, s)$ is separated from ε or more from each facet corresponding to a monomial. So the center ball $(\boldsymbol{p}, s)$ of radius ε is in $\mathcal{D}(f)$. $\mathcal{D}(f)$ and is of dimension $d + 1$.

□

One of the problems of applying neural networks is the expressiveness of the selected family of functions. One method to study these families of functions is to consider the dome and associated hypersurfaces.

3 Hypersurfaces and linear regions

Definition 3.1 (Hypersurface)**.** For a tropical polynomial of the form:

$$f(\boldsymbol{x}) = c_1 \boldsymbol{x}^{\alpha_1} \oplus ... \oplus c_d \boldsymbol{x}^{\alpha_d}$$

The hypersurface is defined by:

$$\mathcal{T}(f) := \left\{ \boldsymbol{x} \in \mathbb{R}^d \mid c_i \boldsymbol{x}^{\alpha_i} = c_j \boldsymbol{x}^{\alpha_j} = f(\boldsymbol{x}),\ \alpha_i \neq \alpha_j \right\}$$

Intuitively it corresponds to the points of change of linearity, or a shift from one monomial to another.

Thus the tropical hypersurface divides the space into regions where the neural network behaves as a linear function.

From a machine learning point of view, we want to be able to express the complexity of a neural network, which corresponds to the number of linear regions that can be represented.

It is clear that these notions are related, indeed in a network of neurons, a change in linearity indicates the presence of 2 linear regions.

In consequence, want to focus more specifically on these tropical objects.

3.1 Transformation of hypersurfaces in a neural network

As we have explained, the transformations in a network of neurons are two in number: linear combination and activations: Relu ie passage to max in our case.

The linear combination poses a problem for the study of hypersurfaces, indeed :
Let f_1, f_2 be tropical polynomials of $\mathbb{R}^d \to \mathbb{R}^d$, writing $\sum$ the tropical sum, $f_1(\boldsymbol{x}) = \sum_i \boldsymbol{x}^{\boldsymbol{a_i^1}}$, $f_2(\boldsymbol{x}) = \sum_j \boldsymbol{x}^{\boldsymbol{a_j^2}}$, what to do with the classical sum $f_1(\boldsymbol{x}) + f_2(\boldsymbol{x}) = \sum_{i,j} \boldsymbol{x}^{\boldsymbol{a_i^1}+\boldsymbol{a_j^2}}$? To my knowledge, there is no theorem for directly linking $\mathcal{T}(f_1), \mathcal{T}(f_2)$ to $\mathcal{T}(f_1 + f_2)$.

One track one might explore is the development of these max. Noting that without prejudice of any simplification, the tropical sum of two polynomials composed of p and q monomials gives a tropical polynomial composed of p $timesq$ monomials. In a neural networks, however, many classical additions are made: :

Note that for a linear transformation of a layer of a neural network $\sigma : \mathbb{R}^l \to \mathbb{R}^t$, for l and t integer, applied to tropical polynomials composed of p monomials, then the image of these tropical polynomials will be tropical polynomials of p^l monomials! The complexity is exponential, we are very quickly limited in the choice of neural networks that this method can describe.

For example, a network with latent spaces of dimensions $2 * 4 * 4 * 1$ can produce more than 10^9 monomials, which poses practical memory problems. Simplifications are possible (monomials can be removed at each stage because they are never "attained", but this reduction is not significant.

One of the interesting ideas on the training of neural networks is that during a successful training, with a suitable architecture, these monomials, and therefore the portion of the network that is not used is relatively small. Starting from this idea of silent monomials and neurons, we can then better design neural networks so as not to over-fit, and learn more about the training i.e the optimization process of a network. This is an interesting research track that brings into play the combinatorics resulting from convex optimization and geometry.

The approach by study of hypersurfaces seems inadequate. This is why many authors like Chang and Al. [16] or Maragos and Al. [3] have chosen to study an object that can be related by duality to the hypersurface: the Newton's polytope.

3.2 Link between hypersurface and Newton's polytope

We now want to prove the link between hypersurface and the Newton's polytope.

Proposition 3.1. *The tropical tropical hypersurface $\mathcal{T}(f)$ coincides with the image of the 2 codimension skeleton of the $\mathcal{D}(f)$ dome in $\mathbb{R}^d$ following the orthogonal projection that removes the last dimension.*

Intuitively at each vertex of the projection of Newton's polytope, we will be able to match a convex cell in $\mathbb{R}^d$ i.e a zonotope where the neuron network will be linear. This bijection is very interesting for the enumeration of linear regions.

Proof. Each facet of the polyhedron $\mathcal{D}(f)$ corresponds to a tropical monomial. A dot $\boldsymbol{p}$ of $\mathbb{R}^d$ belongs to the hypersurface if and only if there are two monomials of f which are equal and are worth $\boldsymbol{p}$ $f(\boldsymbol{p})$. It i;plies that $\boldsymbol{p}, f(\boldsymbol{p})$ in two facets of $\mathcal{D}(f)$.

□

Definition 3.2 (Newton's polytope)**.** The Newton's polytope (extended) of a tropical polynomial f is the convex hull of the points representing its monomials:

$$\mathcal{N}(f) = conv\left\{(\boldsymbol{c_i}, b_i) \in \mathbb{R}^d \times \mathbb{R}, i = 1, ..., d)\right\}$$

$$\mathcal{N}(f) = \left\{\sum_{i=1}^{d} \lambda_i \dot{(}\boldsymbol{c_i}, b_i)| \sum_{i=1}^{d} \lambda_i = 1, 0 \leq \lambda_i\right\}$$

We also define the subdivision induced by f, $\delta(f)$, writing UF for the upper convex hull:

$$\delta(f) = \left\{\pi(\boldsymbol{p}) \in \mathbb{R}^d \mid \boldsymbol{p} \in UF(\mathcal{N}(f))\right\}$$

We start by defining the Minkowski sum, which is widely used in geometry :

Definition 3.3 (Minkowski sum)**.** The Minkowski sum of two sets P_1 and P_2 in $\mathbb{R}^d$ is the set:

$$P_1 \oplus P_2 := \left\{\boldsymbol{x_1} + \boldsymbol{x_2} \in \mathbb{R}^d \mid \boldsymbol{x_1} \in P_1, \boldsymbol{x_2} \in P_2\right\}$$

Where + is the sum component by component.

Proposition 3.2. *Let $f_1, f_2, ..., f_d$ be tropical polynomials of $\mathbb{R}^d$ to$\mathbb{R}$, then writing $\bigoplus$ for the Minkowski sum and $\sum$ for the tropical sum:*

$$\mathcal{T}(\sum_{i=1}^{d} f_i) = \bigcup_{i=1}^{d} \mathcal{T}(f_i)$$

$$\mathcal{N}(\sum_{i=1}^{d} f_i) = \bigoplus_{i=1}^{d} \mathcal{N}(f_i)$$

Proof. The first proof is in [8] lemma 4.7.

For the second, it is sufficient to note that for 2 polynomials :

$$f_1(\boldsymbol{x}) := \sum_{i \in I} \boldsymbol{b_i} + \boldsymbol{x}^{\boldsymbol{c_i}},\ f_2(\boldsymbol{x}) := \sum_{j \in J} \boldsymbol{b_j} + \boldsymbol{x}^{\boldsymbol{c_j}}$$

$$f_1(\boldsymbol{x}) + f_2(\boldsymbol{x}) = \sum_{i \in I,\, j \in J} (\boldsymbol{b_i} + \boldsymbol{x}^{\boldsymbol{c_i}}) + (\boldsymbol{b_j} + \boldsymbol{x}^{\boldsymbol{c_j}})$$

□

Definition 3.4 (Zonotope)**.** A zonotope is a polytope of $\mathbb{R}$ that can be written as a Minkowski sum of segments.

We are interested in the representation by tropical polynomials of a network. In [16], Zhang et al. proves that we can associate with each neural network with activation **Relu** a quotient of tropical polynomials f and g of the form :

$$\frac{f(x_1, ..., x_d)}{g(x_1, ..., x_d)}$$

.

Definition 3.5 (Dual of a polytope)**.** To each polytope we can associate a dual polytope, the idea is that on each side we can associate a vertex and reciprocally.

When we tried to apply those transformations on the hypersurfaces layer-by-layer, we saw that it lea to a dead end.

On the other hand, the study of Newton's polytope (extended) is more promising :

Proposition 3.3. *Let f be a tropical polynomial of $\mathbb{T}[x_1, ..., x_d]$ and a an integer :*

$$\mathcal{N}(f^a) = a\,\mathcal{N}(f)$$

Proposition 3.4. *Let f, g be two tropical polynomials of $\mathbb{T}[x_1, ..., x_d]$:*

$$\mathcal{N}(max(f, g)) = Conv(\mathcal{V}(\mathcal{N}(f)) \cup \mathcal{V}(\mathcal{N}(G))$$

Proof. Knowing that for $f : \boldsymbol{x} \to \sum b_i + \boldsymbol{c_i x}$;

$$\mathcal{N}(f) = Conv\left\{(\boldsymbol{c_i}, b_i) \in \mathbb{R}^d \times \mathbb{R}, i = 1, ..., d)\right\}$$

□

We deduce the layer-to-layer transformations of the (extended) Newton's polytope:

Proposition 3.5. *Let $A^{(1)} \in \mathbb{Z}^{n_2 \times n_1}$ and $b^{(1)} \in \mathbb{R}^{n_2}$ the weight matrices of the first layer, $A^{(1)}$ is decomposed into 2 positive coefficient matrices : $A^{(1)} = A^{(1)}_+ - A^{(1)}_-$*

We obtain the Newton's polytopes associated with the transformations: $\mathcal{N}(f_i^{(l)}), \mathcal{N}(g_i^{(l)})$ and $\mathcal{N}(h_i^{(l)})$ from layer to layer applying the transposition of tropical transformations on the Newton's polytope matrices :

The first polytopes are obtained directly:

(i) $\mathcal{N}(g_i^{(1)})$ and $\mathcal{N}(h_i^{(1)})$ are point

(ii) $\mathcal{N}(f_i^{(1)})$ is a line

(iii) $\mathcal{N}(g_i^{(2)})$ and $\mathcal{N}(h_i^{(2)})$ and the next are zonotopes defined by :

At step $l \geq 1$:

$$\mathcal{N}(f_i^{(l)}) = Conv[\mathcal{N}(g_i^{(l)}) \cup \mathcal{N}(h_i^{(l)})]$$

And Newton's polynomials are weighted Minkowski sums :

$$\mathcal{N}(g_{i+1}^{(l)}) = \sum_{j=1}^{n_l} a_{i,j}^{-} \mathcal{N}(f_j^{(l)}) + \sum_{j=1}^{n_l} a_{i,j}^{+} \mathcal{N}(g_j^{(l)})$$

$$\mathcal{N}(h_{i+1}^{(l)}) = \sum_{j=1}^{n_l} a_{i,j}^{+} \mathcal{N}(f_j^{(l)}) + \sum_{j=1}^{n_l} a_{i,j}^{-} \mathcal{N}(g_j^{(l)}) + b_i * e_{d+1}$$

With $a_{i,j}$ the terms of the matrix $A^{(l+1)} \in \mathbb{Z}^{n_{l+1} \times n_l}$ and b_i for $b^{(l+1)} \in \mathbb{R}^{n_{l+1}}$ the weight matrices associated to the $l+1$ th layer and e_{d+1} the vector unitary of the base with a 1 in $d+1$ th position : $(0, .., 0, 1)$.

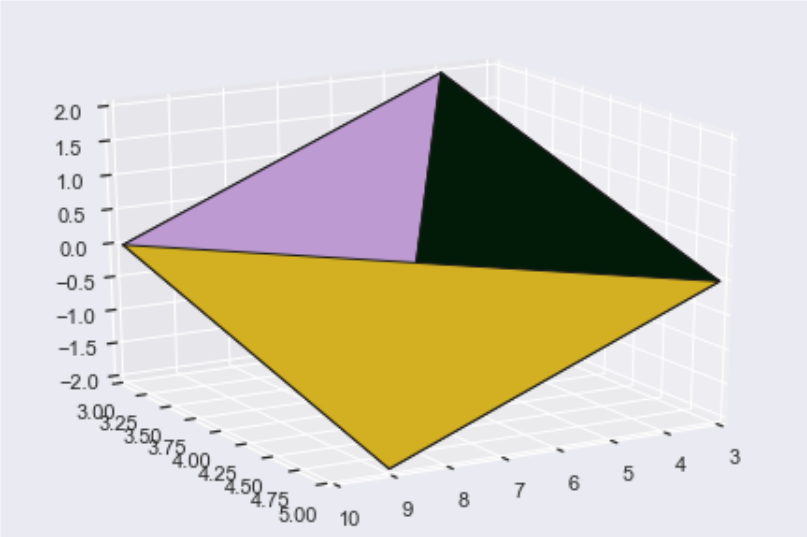

Figure 6: The convex polytope $\mathcal{N}(g^{(2)})$

3.3 Application

The goal is to give a representation of the linear regions in the Relu layers and a representation of the associated tropical varieties.

Consider for example a two-layer network, $\nu : \mathbb{R}^2 \to \mathbb{R}$ with $n_0 = 2$ inputs, $n_1 = 5$ for the first layer and $n_2 = 1$ as the dimension of the output. With the previous notations:

$$\boldsymbol{y} = \nu^{(1)}(\boldsymbol{x}) = \sigma \circ \rho^{(1)}(\boldsymbol{x}) = max\left\{ \begin{pmatrix} -1 & 1 \\ 3 & 0 \\ 1 & -2 \\ 2 & 1 \\ 3 & -3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 2 \\ -1 \\ 2 \\ 0 \\ -2 \end{pmatrix}, 0 \right\}$$

$$\nu^{(2)}(\boldsymbol{y}) = max(-3y_1 + 2y_2 + y_3 - 2y_4 - y_5, 0)$$

The network satisfies criteria $(a) - (c)$, by 3.5 so we can apply our tropical transposition.

We are looking to write $\nu^{(1)} = f^{(1)} \oslash g^{(1)}$ Under the form $f^{(1)}(\boldsymbol{x}) = h^{(1)}(\boldsymbol{x}) \oplus g^{(1)}(\boldsymbol{x})$ where h and g are tropical polynomials. To do this, separate:

$$\nu^{(1)} = max(A\boldsymbol{x} + \boldsymbol{b}, \boldsymbol{0}) = max(A_+^{(1)}\boldsymbol{x} - A_-^{(1)}\boldsymbol{x} + \boldsymbol{b}, \boldsymbol{0})$$

then

$$g^{(1)}(\boldsymbol{x}) = \begin{pmatrix} x_1 \\ 0 \\ x_2^2 \\ 0 \\ x_2^3 \end{pmatrix} \quad h^{(1)}(\boldsymbol{x}) = \begin{pmatrix} 2 \odot x_2 \\ -1 \odot x_1^3 \\ 2 \odot x_1 \\ x_1^2 x_2 \\ -2 \odot x_1^3 \end{pmatrix}$$

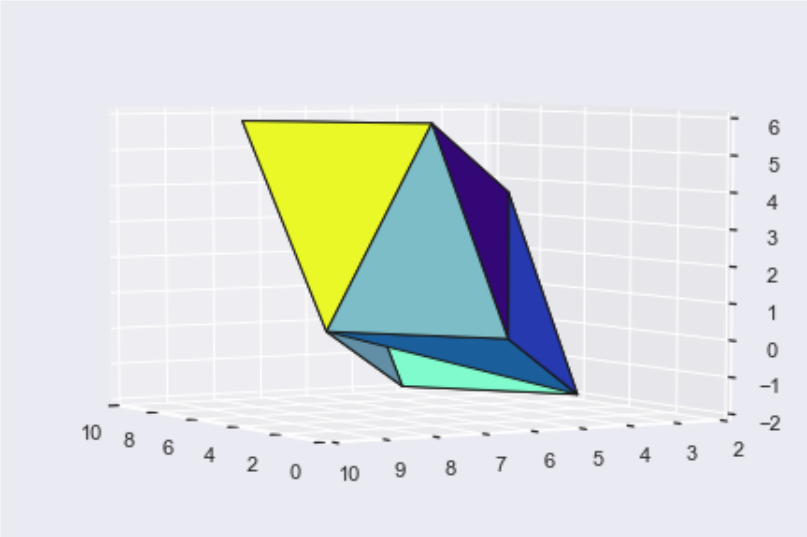

Figure 7: The convex polytope $\mathcal{N}(h^{(2)})$

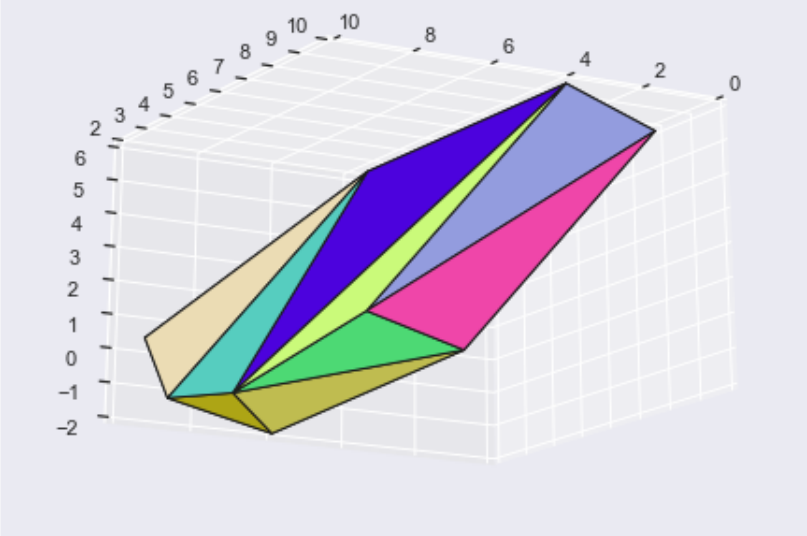

Figure 8: The convex hull of Newton's polytope $\mathcal{N}(f^{(2)})$

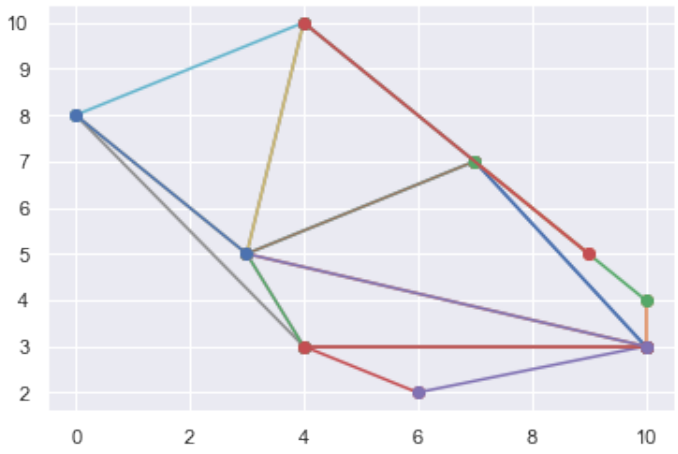

Figure 9: Projection by π of the convex hull of $\mathcal{N}(f^{(2)})$

In the same way we try to write $f^{(2)}(\boldsymbol{x}) = h^{(2)}(\boldsymbol{x}) \oplus g^{(2)}(\boldsymbol{x})$, by defining $\boldsymbol{z} = g^{(1)}$ and $\boldsymbol{y} = f^{(1)}$, and we want $\nu^{(2)} = f^{(2)} \oslash g^{(2)}$. The layer $\nu^{(2)}$ can be written :

$$g^{(2)}(\boldsymbol{x}) = y_1^3 \odot z_2^2 \odot z_3 \odot y_4^2 \odot y_5 = (2 \odot x_2 \oplus x_1)^3 \odot 0^2 \odot x_2 \odot (x_1^2 x_2 \oplus 0)^2 \odot (-2 \odot x_1^3 \oplus x_2^3)$$
$$h^{(2)}(\boldsymbol{x}) = z_1^3 \odot y_2^2 \odot y_3 \odot z_4^2 \odot z_5 = x_1^3 \odot (-1 \odot x_1^3 \oplus 0)^2 \odot (2 \odot x_1 \oplus x_2^2) \odot 0^2 \odot x_2^3$$

Now consider the tropical varieties associated with the neural network. We write $f^{(1)} = (f_1^{(1)}, f_2^{(1)}, f_3^{(1)}, f_4^{(1)}, f_5^{(1)})$ and do the same for $g^{(1)}$ and $h^{(1)}$. We note that $g_j^{(1)}$ and $h_j^{(1)}$ are tropical monomials (of the form $cx_1^{a_1} x_2 a_2$). Associated Newton's polytopes, $\mathcal{N}(g_j^{(1)})$ and $\mathcal{N}(h_j^{(1)})$ are then points in $\mathbb{R}^3$.
As $f^{(1)} = h^{(1)} \oplus g^{(1)}$, it follows $\mathcal{N}(f_j^{(1)})$ which is the convex hull of two points in $\mathbb{R}^3$, a line.

The lines $\mathcal{N}(f_j^{(1)})$ and the points $\mathcal{N}(g_j^{(1)})$ combine to form the varieties $\mathcal{N}(g_j^{(2)})$ and $\mathcal{N}(h_j^{(2)})$, writing $+$ the Minkowski sum, transposing the previous equations :

$$\mathcal{N}(g^{(2)}) = 3\mathcal{N}(f_1^{(1)}) + 2\mathcal{N}(g_2^{(1)}) + \mathcal{N}(g_3^{(1)}) + 2\mathcal{N}(f_4^{(1)}) + \mathcal{N}(f_5^{(1)})$$
$$\mathcal{N}(h^{(2)}) = 3\mathcal{N}(g_1^{(1)}) + 2\mathcal{N}(f_2^{(1)}) + \mathcal{N}(f_3^{(1)}) + 2\mathcal{N}(g_4^{(1)}) + \mathcal{N}(g_5^{(1)})$$

Finally $\mathcal{N}(f^{(2)})$ is the convex hull of the union of $\mathcal{N}(g^{(2)})$ and $\mathcal{N}(h^{(2)})$. The tropical hypersurface $\mathcal{T}(\nu)$ is then the skeleton of the dual of the projection removing the last coordinates of the upper faces of $\mathcal{N}(f^{(2)})$ by the proposition 3.1.

4 Computational geometry

In order to be able to represent and compute those surfaces, it is necessary to be able to determine the convex hull of points. Another difficulty of the study of neural networks by transformations in the tropical world is the possible large number of terms in the Minkowski sum.

The basic concept of computer representation is first discussed: triangulation.

4.1 Triangulation of polygons

In order to draw, represent or store surfaces or geometric objects, we use triangulation [15] and the concept of simplex.

Definition 4.1 (simplex)**.** A n simplex is the convex hull of a set of $n+1$ points. When $n+1$ points form a linearly independent set it is called a affine simplex.
A k face is the convex hull of k points.
A simplicial complex K is a set of affine simplexes such that if $\sigma \in K$, all faces of σ are in K and for two elements σ_1, σ_2 of K then $\sigma_1 \cap \sigma_2$ is empty or is both a face of σ_1 and a face of σ_2.
We call support of K, a simplicial complex, the following construction : $|K| = \cup \{\sigma \mid \sigma \in K\}$

Definition 4.2 (triangulation)**.** We say that a set X is triangulable if there is a homeomorphism between X and a simplicial complex support $|K|$. The Homeomorphism and K are called triangulation of X.

Before going further, it is interesting to mention a triangulation method that is widely used : Delaunay's triangulation. To build a geometric design, this triangulation provides a grid that 'maximizes' the minimum angle. This triangulation is used to find accuracy of interpolations.[15]

Definition 4.3 (Delaunay's triangulation)**.** For a finite set of S points, a simplex is called Delaunay if its vertices are in S and no point of S is inside the circumscribed sphere associated with these vertices.
A triangulation is called Delaunay if all his simplexes are Delaunay.

There are several algorithms for triangulation in 3d: local enhancement, incremental construction ... [11]. It is also well known that the Delaunay triangulation of a discrete set of points is the dual graph of the Voronoi diagram associated with P [15].

Let us now consider the calculation of the convex hull and the upper convex hull.

4.2 Convex hull and upper convex hull

We will have to compute the convex hull as well as the upper convex hull to be able to visualize the hypersurfaces.
In this part one chooses to present 2 classical methods to determine the convex hull of a set of points in 3d. Then is proposed a method to find the upper convex hull.

4.2.1 Incremental method

The idea is to calculate the convex hull on a growing set of points.

The algorithm can be written for n points $p_1 ... p_n$ of a set S, to find H the convex hull:

- initialize $H = Conv(p_1, p_2, p_3, p_4)$, H is a collection of faces
- pour $i = 5$ à $i = n$:
  - for each face f of H :
    - compute the volume v of the tetrahedron formed by (f, p_i)
    - if $v < 0$ the tetrahedron is visible
  - if no face is visible :
    - p_i is an interior point, to be ignored for the convex hull
  - if not :
    - find the edges of all visible faces
    - for each e edge, construct the (e, p_i) face and add it to H
    - for each visible face f, delete f from H

A 'positive' volume corresponds to a point behind 'the face, so external to H. The complexity is $O(nlog(n)$ and $O(n^2)$ in the worst case [2].

4.2.2 Quickhull Method

The Quickhull method is commonly used to calculate the convex hull, especially when the dimension of the space is greater than three. The three-dimensional algorithm takes the ideas of incremental construction (also called Beneath and Beyond) to be able to apply the Quickhull 2d method to a larger dimension [1].

The Quickhull method in 2d is the following for a set of points S :

- Initialization: draw the line d connecting the two extreme points along the x axis
- We consider the same d and S as before:
  - separate the S point in two subsets S_1, S_2 according to their placement relative to the line d
  - for each of the two subsets, find the farthest point on the line extending d
  - draw the triangle that connects this point and the ends of d
  - exclude from the calculation the points inside the triangle: it remains the set S^o
  - repeat the operation with the two new lines d_1 and d_2 from the drawing of the triangle and the set S^o
- when the points of S^o coincide with all the segments, the convex hull is obtained

The 3D method follows the same reasoning, but one must be careful to take into account the vertices, edges, and faces correspondences in order to be able to update vertices, edges and faces when one modifies the convex hull.
The algorithm can be written:

- Initialization: begin with a plane characterized by 3 points, we find the furthest point from the plane of the triangle, with these 4 points we form a first convex hull H, we can exclude from the calculation the internal points that are not 'visible'

- Select the point furthest from the points already included. Add a new vertex to H as follows :
  - Calculate the horizon: the edges that can create a positive volume, that is to say, which can 'see' the point: start with the nearest face and then to its neighbors
  - Same process as with the incremental method
  - Merge the coplanar faces

The complexity is in $O(nlog(n))$ in the general case.

4.3 Minkowski sum of convex polytopes

Our study of the hypersurface of decision pushes us to study the various transformations of the Newton's polytopes associated with the network.

As we saw in the previous part, the step that creates the most linear regions is the Minkowski sum of several polytopes.

We must then find a way to effectively represent the resulting polytopes.

The main difficulties of the implementation are the number of vertices in the Minkowski sums and the choice of the representation.

An important point is indeed the representation of Minkowski sums of convex polytopes: lattices, vertices, etc. We choose to keep the representation by vertices, we then have a simplified version through the convex hull.

One of the possible simplifications to reduce the number of vertices before passing to the convex hull characterizing the convex polytope is to verify if the next vertex in the Minkowski sum is not "inside the structure", ie to verify that it is not in the convex hull of vertices already obtained.

To test this hypothesis [5] tests the existence of a separating hyperplane.

4.4 Polyhedra and polytopes

Definition 4.4 (Polyhedra)**.** A polyhedron P in $\mathbb{R}^d$ is a convex set, intersection of a finite number of half spaces $H(A, b)$ with $A \in \mathbb{R}^d, b \in \mathbb{R}$:

$$H(A,b) = \left\{ \boldsymbol{x} \in \mathbb{R}^d, \sum_i^d a_i x_i \leq b \right\}$$

$$P = \cap_{j=1}^n H(A_j, b_j)$$

Proposition 4.1 (Minkowski Weyl theorem)**.** *A polytope is a bounded polyhedron.*

This theorem links convex hulls and the half spaces, and shifts the study from geometry to optimization.

5 Structure of polytopes

The aim of this part is to better understand the polytope object, the Weyl Minkowski theorem and the link between convex polytopes and triangulation.

5.1 Introduction on optimization problems

Combinatorial optimization aims at optimizing an objective function subject to certain constraints. We are particularly interested in the systems of equations in $\mathbb{R}^d$ under constraints,of the form :

$$i \in \{1, ..., n\}, \sum_{j=1}^{d} a_{ij} x_j \geq b_i$$

We are interested in the existence of a solution to this system.

One of the best-known ways to answer this question is Farkas's lemma, which is based on the polyhedra generated by these inequalities:

Proposition 5.1 (Non-homogeneous Farkas Lemma)**.** *In $\mathbb{R}^d$, a system of inequalities, ${\boldsymbol{a_1}}^T \boldsymbol{x} \geq b_1, ..., {\boldsymbol{a_n}}^T \boldsymbol{x} \geq b_n$ has no solution if and only if $(\lambda_i)_n$ real positives such as:*

$$\sum_{i=1}^{n} \lambda_i \boldsymbol{a_i} = \boldsymbol{0}$$

$$\sum_{i=1}^{n} \lambda_i b_i = 1$$

Proposition 5.2 (Homogeneous Farkas Lemma)**.** ${\boldsymbol{a_1}}^T \boldsymbol{x} \geq 0, ..., {\boldsymbol{a_n}}^T \boldsymbol{x} \geq 0$ *implies* $\boldsymbol{c}^T \boldsymbol{x} \geq 0$ *i and only if there are $(\lambda_i)_n$ real positives such as :*
$\sum_{i=1}^{n} \lambda_i \boldsymbol{a_i} = \boldsymbol{c}$

We will prove those theorems a little later. Let's go back to the polytopes.

5.2 Polytopes

Proposition 5.3 (Hyperplan of separation)**.** *Let X be a compact convex set of $\mathbb{R}^d$, and z an element outside X, then there are reals $(a_i)_d$ and a strictly positive real ε such that:*

$$\forall \boldsymbol{x} \in X, \sum_{i=1}^{d} a_i x_i \leq \varepsilon$$

$$\sum_{i=1}^{d} a_i z_i \geq \varepsilon$$

the objects that will be studied with this property will be the polytopes, generated by a finite set of points. They are therefore compact. We realize that without the closing property the theorem is not verified.

Proof. The closing property allows us to state that there is at least one point $\boldsymbol{x}$ of X as $||\boldsymbol{x}-\boldsymbol{z}|| = min_{\boldsymbol{x}\in X} = ||\boldsymbol{x}-\boldsymbol{z}|| > 0$. The convexity of X forces its uniqueness: indeed if it is not unique let's call $\boldsymbol{y} \in X$ another point where this minimum is reached.
We are now looking at the line segment $[\boldsymbol{xy}]$. It is included in X per convexity and the middle of this segment will be at a distance of $\boldsymbol{z}$ strictly lower than $||\boldsymbol{x}-\boldsymbol{z}||$, which contradicts the minimality of the distance of $\boldsymbol{z}$ to X.

We now consider the center ball $\boldsymbol{z}$ and radius $||\boldsymbol{x}-\boldsymbol{z}||$. We show that every X is on the same side of the hyperplane, characterized by the vector $\vec{n}$.
Indeed, let's suppose $\boldsymbol{y}$ a point of the same side as the ball of the hyperplane. $\overrightarrow{n} \cdot \overrightarrow{\boldsymbol{y}} > 0$ for example. $\boldsymbol{x}$ being on this hyperplane, then $\overrightarrow{\boldsymbol{y}-\boldsymbol{x}} \cdot \overrightarrow{\boldsymbol{n}} > 0$. We deduce that the line segment $[\boldsymbol{xy}]$ intersects the ball in at least one point outside of the hyperplane, written $\boldsymbol{u}$.

By convexity the line segment $[\boldsymbol{xy}]$ is in X then $\boldsymbol{u}$ in X and $||\boldsymbol{u}-\boldsymbol{z}|| = ||\boldsymbol{x}-\boldsymbol{z}||$. This contradicts the uniqueness of $\boldsymbol{x}$.We obtain the result by translating the hyperplane of $||\boldsymbol{x}-\boldsymbol{z}||/2$ along $\vec{\boldsymbol{n}}$. □

This theorem will prove useful when performing the Minkowski sum of convex polytopes on the vertices, to know if a potential vertex is already in the described convex region as shown in [5]. We can thus reduce the number of vertices in the Minkowski sum.

Let's now discuss three theorems about the combinatorics of convex sets :

Proposition 5.4 (Caratheodory)**.** *In $\mathbb{R}^d$ any element $\boldsymbol{x} \in conv(S) = \left\{\sum_i^k \lambda_i \boldsymbol{x_i}, \sum \lambda_i = 1, \lambda_i > 0\right\}$ can be written as $\boldsymbol{z} = \sum_i^p \lambda_i \boldsymbol{x_i},\ p \leq d+1$.*

Proof. If $k \leq d+1$, just apply the definition of the convex hull.
When $k > d+1$, define $\boldsymbol{x_i}' = (1, \boldsymbol{x_i})$ on $\mathbb{R}^{d+1}$, it follows $k > d+1$ vectors in $\mathbb{R}^{d+1}$, therefore exist μ_i reals not all equals to zero such as:

$$\sum \mu_i \boldsymbol{x_i}' = 0$$

The first dimension being only ones follows $\sum \mu_i = 0$, and there is at least non zeros μ_+ and μ_- of opposing signs. Therefore we can replace one of the $\boldsymbol{x_i}$ with a linear combination of the other elements. The result is obtained by recurrence.

□

Proposition 5.5 (Helly)**.** *Let in $\mathbb{R}^d$, C a finite collection of convex sets such as any $d+1$ sets of elements of C has a non-empty intersection, then the intersection on all the sets in C is not empty.*

Proof. Let's start by showing the theorem for a collection of $d+2$ sets.

Let $d+2$ convex sets, bu the hypothesis exists at least one point inside the intersections of the $d+1$ sets that we write $\boldsymbol{a_i},\ i = 1...d+2$ and Δ_i the convex hull associated to $d+1$ of those points :

$$\Delta_i = conv(\boldsymbol{a_1}, ..., \boldsymbol{a_{i-1}}, \boldsymbol{a_{i+1}}, ...\boldsymbol{a_{d+2}})$$

Let the function f of distance to the convex hulls : $f(\boldsymbol{x}) = max_i(d(\boldsymbol{x}, \Delta_i))$

Its minimum on $conv(\boldsymbol{a_1}, ..., \boldsymbol{a_{d+2}})$, is attained on $\boldsymbol{z}$. The goal is to show $f(\boldsymbol{z}) = 0$, which is a proof of the existence of a point belonging to the $d+2$ sets.

By application of the Caratheodory theorem, it is known that $\boldsymbol{z}$ belongs to one of the Δ_i, writing $\boldsymbol{z} \in \Delta_1$ without loss of generality.

We are interested in the line segment $[\boldsymbol{z a_1}]$, and more precisely to $f(\boldsymbol{x}_\theta) = f((1-\theta)\boldsymbol{z} + \theta \boldsymbol{a_1})$ for $\theta \in [0, 1]$. By convexity $[\boldsymbol{z a_1}]$ is in $conv(\boldsymbol{a_1}, ..., \boldsymbol{a_{d+2}})$.

Reminding that $\boldsymbol{a_1} = \bigcap_{j=2}^{d+2} \Delta_j$.

Let $\pi_i(\boldsymbol{z})$ the projection of $\boldsymbol{z}$ on Δ_i, by triangular inequality, follows for any i:

$$d(\boldsymbol{x}_\theta, \Delta_i) \leq d((1-\theta)z, \Delta_i) + d(\theta a_1, \Delta_i) = (1-\theta)d(z, \Delta_i) \leq (1-\theta)f(z)$$

pour tout i = 2...d+2.

Therefore for any $\theta \in [0, 1]$:

$$f(\boldsymbol{z}) \leq f(\boldsymbol{x}_\theta) \leq max[d(\boldsymbol{x}_\theta, \Delta_1), (1-\theta)f(\boldsymbol{z})]$$

Inequality is simplified by $f(\boldsymbol{z}) \leq d(\boldsymbol{x}_\theta, \Delta_1)$ by making θ converge toward 0, the theorem is obtained for d+2 sets.

Note that this reasoning is valid for any size of collections of convex sets, and shows that the non-empty intersection on n + 1 sets implies the non-empty intersection on n + 2 sets, for $n > d$.

The theorem is proved by recurrence. □

Proposition 5.6 (Radon)**.** *In $\mathbb{R}^d$ any set A containing at least d+2 points admits a partition which two parts X,Y verify $conv(X) \cap conv(Y) \neq \emptyset$.*

Proof. Let $A = \{\boldsymbol{a_1}, ..., \boldsymbol{a_{d+2}}\}$, and the λ_i *reals*, we consider the system of equations in λ :

$$\sum_{j=1}^{d+2} \lambda_j \boldsymbol{a_j} = 0$$

$$\sum_{j=1}^{d+2} \lambda_j = 0$$

Having d + 1 equations for d + 2 unknowns, there is a non-null solution. We notice

$$I_1 = \{\lambda_i,\ \lambda_i > 0\}$$

$$I_2 = \{\lambda_i,\ \lambda_i \leq 0\}$$

Choosing the partition $X = \{\boldsymbol{a_i}, \in I_1\}$, $Y = \{\boldsymbol{a_i}, \in I_2\}$

Then :

$$\frac{\sum_{i \in I_1} \boldsymbol{a_i} \lambda_i}{\sum_{i \in I_1} \lambda_i} = \frac{\sum_{i \in I_2} \boldsymbol{a_i} \lambda_i}{\sum_{i \in I_2} \lambda_i}$$

Which corresponds to a point of the convex hull of X which is also in the convex hull of Y.

□

5.3 Weyl Minkowski theorem

Weyl Minkowski theorem states that every polytope is a bounded polyhedron.

To show this theorem we are interested in the constraint matrix, and its rank.

In $\mathbb{R}^d$ optimization problem constraints are :

$$i \in \{1, ..., n\}, \sum_{j=1}^{d} a_{ij} x_j - b_i \geq 0$$

That can be summarized to:

$$i \in \{1, ..., n\}, c_i(\boldsymbol{x}) \geq 0$$

S the space described by these constraints, $S \neq \emptyset$.

We define three notions of "corner points" of S :

Definition 5.1 (vertex of S)**.** x is a vertex of S if there exists $\boldsymbol{c} \in \mathbb{R}^d$ as : $\forall \boldsymbol{y} \in S \setminus \{\boldsymbol{x}\}$, $\boldsymbol{c}^T \boldsymbol{x} > \boldsymbol{c}^T \boldsymbol{y}$.

The idea is that it is not possible to go further in the corner point direction direction.

Definition 5.2 (extreme point)**.** Let S the convex set associated to these constraints, non-trivial. $\boldsymbol{x}$ is an extreme point of S if it can not be written as a combination of two other points of S.

This definition hints at the convexity construction of S.

Definition 5.3 (Constraints matrix)**.** Let $A_{\boldsymbol{x}}$ the constraints matrix in $\boldsymbol{x}$, of shape $n \times d$:

$$A_{\boldsymbol{x}} = \left\{ \boldsymbol{a_i} | \boldsymbol{a_i}^T \boldsymbol{x} = b_i \right\} = \{ \boldsymbol{a_i} | i \in I_{\boldsymbol{x}} \}$$

with

$$I_{\boldsymbol{x}} = \left\{ i | \boldsymbol{a_i}^T \boldsymbol{x} = b_i \right\}$$

All these definitions are related.

Proposition 5.7. *Let S be a polyhedron, the following three propositions are equivalent:*

(i) $\boldsymbol{x}$ *is a vertex of* S

(ii) $\boldsymbol{x}$ *is a corner point of* S

(iii) $A_{\boldsymbol{x}}$ *is of rank* d

Proposition 5.8. $\boldsymbol{x}$ *is an extreme point of* S *if and only if the rank of* $A_{\boldsymbol{x}}$ *is* d.

The idea is to show that $\boldsymbol{x}$ is at the intersection of hyperplanes that correspond to linearly independent constraints.

Proof. $(i) \implies (ii)$: by contradiction, let $\boldsymbol{x}$ a vertex of S associated to the direction $\boldsymbol{c}$, suppose there exists $\alpha \in [0, 1]$ and $\boldsymbol{y}, \boldsymbol{z} \in S$ as : $\boldsymbol{x} = \alpha \boldsymbol{y} + (1 - \alpha)\boldsymbol{z}$. Multiplying the inequality par $\boldsymbol{c}^T$ we deduce that $\boldsymbol{x} = \boldsymbol{y} = \boldsymbol{z}$.

$(ii) \implies (iii)$: If $\boldsymbol{x}$ is an extreme point and the rank of $A_{\boldsymbol{x}}$ is inferior to d, there exists a vector $\boldsymbol{v}$ non trivial as $A_{\boldsymbol{x}} \boldsymbol{v} = 0$. Noticing that $\boldsymbol{a}^T \boldsymbol{x} - b_i > 0$ for $i \notin I_{\boldsymbol{x}}$, and $\boldsymbol{a}^T \boldsymbol{v} = 0$ for $i \in I_{\boldsymbol{x}}$ therefore there is a $\varepsilon > 0$ that verifies $\boldsymbol{x} + \varepsilon \boldsymbol{v} \in S$ and $\boldsymbol{x} - \varepsilon \boldsymbol{v} \in S$. It contradicts the definition of the extreme point.

$(iii) \implies (i)$: Picking $\boldsymbol{c} = \sum_{I_{\boldsymbol{x}}} \boldsymbol{a_i}$, implies $\boldsymbol{c}^T\boldsymbol{x} = \sum_{I_{\boldsymbol{x}}} \boldsymbol{a_i}^T\boldsymbol{x} = \sum_{I_{\boldsymbol{x}}} b_i$.

Let $\boldsymbol{y} \in S$, then $\boldsymbol{c}^T\boldsymbol{y} = \boldsymbol{a_i}^T\boldsymbol{y} \leq \sum_{I_{\boldsymbol{x}}} b_i = \boldsymbol{c}^T\boldsymbol{x}$. The inequality implies $\forall \boldsymbol{a_i} \in A_{\boldsymbol{x}}, \boldsymbol{a_i}^T\boldsymbol{y} = \boldsymbol{a_i}^T\boldsymbol{x}$, $A_{\boldsymbol{x}}$ being of rank d, then $\boldsymbol{x} = \boldsymbol{y}$ in $\mathbb{R}^d$.

□

From this result is obtained a quick corollary :

Corollary 5.8.1. *A polyhedron has a finite number of extreme points.*

Proof. Each extreme point corresponds to an additional vector in constraint matrix of shape $n \times d$, then there are $\binom{n}{d}$ possibilities to form a rank d matrix. □

Getting closer to Minkowski Weyl theorem with the first implication:

Proposition 5.9. *Let P a bounded polyhedron of vertices $p_1, ..., p_n$ then $P = conv(p_1, ..., p_n)$.*

Proof. Recall that a polyhedron P in $\mathbb{R}^d$ is a convex set, intersection of a finite number of half spaces. By convexity, comes directly $conv(p_1, ..., p_n) \subseteq P$. Let's show that z in P implies z in $conv(p_1, ..., p_n)$. To do this let's reason by induction on $d - rang(A_{\boldsymbol{z}})$.

If $d - rang(A_{\boldsymbol{z}}) = 0$, the previous lemma tells us that $\boldsymbol{z}$ is a vertex.

If $d - rang(A_{\boldsymbol{z}}) > 0$, there is a vector $\boldsymbol{v}$ non trivial as $A_{\boldsymbol{z}}\boldsymbol{v} = 0$. Defining on P compact $k_+ := max\{k | \boldsymbol{z} + k\boldsymbol{v} \in P\}$, $k_- := max\{k | \boldsymbol{z} - k\boldsymbol{v} \in P\}$ attained in $\boldsymbol{x}$ and $\boldsymbol{y}$. Considering $\boldsymbol{x}$, writing i_0 the limiting index, where a new constraint is reached.

Then : $A_{\boldsymbol{z}}\boldsymbol{x} = A_{\boldsymbol{z}}\boldsymbol{z}$ and $\boldsymbol{a_{i_0}}\boldsymbol{x} = b_{i_0}$

Noticing that $A_{\boldsymbol{x}}$ contains the same columns as $A_{\boldsymbol{z}}$ but with $\boldsymbol{a_{i_0}}$. Therefore $rang(A_{\boldsymbol{x}}) > rang(A_{\boldsymbol{x}})$.

By the same reasoning comes $rang(A_{\boldsymbol{y}}) > rang(A_{\boldsymbol{x}})$.

By convexity of S, if the result is valid for $\boldsymbol{y}$ and $\boldsymbol{x}$, then it is for $\boldsymbol{z}$.

The result is obtained by induction.

□

Now is shown the end of the Minkowski Weyl theorem for polytopes.

Proof. let P a polytope $conv(x_1, ..., x_n)$. It remains to prove that P defines a polyhedron.

Choosing P of dimension d. For P different dimensions, the result is obtained by induction.

Let $P^* = \left\{\boldsymbol{y} \,|\boldsymbol{x}^T\boldsymbol{y} \leq 1, \forall \boldsymbol{x} \in P\right\}$. By construction $P^* = \left\{\boldsymbol{y} \,|\boldsymbol{x}_i^T\boldsymbol{y} \leq 1, \forall i\right\}$. Hence P^* is a polyhedron.

P^* is called the polar dual. This transformation is particularly interesting when the polar bidual of a set is it the set of origin.

For $\boldsymbol{y}$ in P^* : $\boldsymbol{x}^T\boldsymbol{y} = \sum \lambda_i \boldsymbol{x}_i^T\boldsymbol{y} \leq \sum \lambda_i = 1$

We therefore deduce that $P^* = \left\{\boldsymbol{y} \,|\boldsymbol{x}_i^T\boldsymbol{y} \leq 1, \forall i\right\}$. P^* is a polyhedron. S being of dimension d then P^* is bounded. Applying a previous theorem, P^* is a polytope. To conclude, it remains to show that $P^{**} = P$.

$P^{**} = \left\{\boldsymbol{y} \,|\boldsymbol{x}^T\boldsymbol{y} \leq 1, \forall \boldsymbol{x} \in P^*\right\}$. Noticing $x_1, ..., x_n$ are not in P^{**}, P^{**} being convex is deduced $P \subseteq P^{**}$.

Assume $x \notin P$, the goal is to show $\boldsymbol{x} \notin P^{**}$, implying $P = P^{**}$.

Considering the hyperplan of separation between $\boldsymbol{x}$ and P, $\boldsymbol{C}$ and δ with $\boldsymbol{C}^T\boldsymbol{z} < \delta \ \forall \boldsymbol{z} \in P$ and $\boldsymbol{C}^T\boldsymbol{x} > \delta$.

Without loss of generality we can transpose in the case where δ is worth 1, as we will justify in the following part on the polar dual.

Therefore: $\boldsymbol{C}^T\boldsymbol{z} < 1\ \forall \boldsymbol{z} \in P$, hence $\boldsymbol{C} \in P^*$. Knowing $\boldsymbol{C}^T\boldsymbol{x} > 1$ then $\boldsymbol{x} \notin P^{**}$.

□

5.4 Duality

What is duality in these spaces? We are familiar with dual spaces in functional analysis.

In this thesis it is chosen to study the dual of linear programming and the polar dual.

5.4.1 Polar duality

The goal of this part is the study of the polar dual and bidual. This transformation will then be used in the proof of Farkas' lemma.

Definition 5.4 (polar dual)**.** In $\mathbb{R}^d$ the polar dual of a set P is defined by $P^* = \left\{\boldsymbol{y}\,|\boldsymbol{x}^T\boldsymbol{y} \leq 1,\ \forall \boldsymbol{x} \in P\right\}$.

We can also define the polar dual for a hyperplane H which does not contain $\mathbf{0}$, if $H = \{\boldsymbol{x}|\boldsymbol{c}\boldsymbol{x} = 1\}$, then the dual of H is $\boldsymbol{c}$.

Definition 5.5 (polar bidual)**.** $P^{**} = \left\{\boldsymbol{c}\,|\boldsymbol{x}^T\boldsymbol{y} \leq 1,\ \forall \boldsymbol{x} \in P \implies \boldsymbol{c}^T\boldsymbol{y} \leq 1\right\}$.

The bidual appears as the intersection of all the half spaces that verify on $P \cup \mathbf{0}$, $\left\{\boldsymbol{x}^T\boldsymbol{y} \leq 1\right\}$.

From the previous part is quickly obtained a property and a theorem :

Proposition 5.10. *P^* is a close convex containing 0.*

Theorem 5.11. *$P = P^{**}$ if and only if P is a closed convex containing $\mathbf{0}$.*

Proof. The ideas have already been implemented in the proof of Minkowski Weyl's theorem. The first inclusion is trivial. The second is based on the theorem of the separation hyperplane previously shown, which is a corollary of Hahn Banach's theorem. The Hahn Banach separation theorem is presented in the appendix.

□

$\mathbf{0}$ must belong to the initial set, therefore the changes necessary about the initial space when it is applied to the polytope in the proof of the Minkowski Weyl theorem.

Here are some remarkable properties:

- The unit ball of l_p^d : $B_p^d = \left\{\boldsymbol{x} \in \mathbb{R}^d; \sum_{i=1}^d |x_i|^p \leq 1\right\}$, if $p \geq 1, \frac{1}{p} + \frac{1}{q} = 1$ verify :

$$(B_p^d)^* = B_q^d$$

- $\mathbf{0} \in int(P) \implies P^*$ bounded, P bounded $\implies \mathbf{0} \in int(P^*)$
- Let P a polytope defined by $P = conv(V), V \in \mathbb{R}^{d\times n}$, then $P^* = \{\boldsymbol{x}|\boldsymbol{x}V \leq \mathbf{1}\}$
- if $P = \{\boldsymbol{x}|A\boldsymbol{x} \leq \mathbf{1}\}$ is bounded, then $P^* = conv(A^T)$

Proof. • follows from Holder inequality

- $B(0,r) \subset P \implies P^* \subset B(0,r)^* = B(0,1/r)$
- let V be a set of n points in $\mathbb{R}^d$, let $\boldsymbol{x} \in \mathbb{R}^d$, $\boldsymbol{x}V \leq \mathbf{1}$ if and only if $\boldsymbol{xy} \leq 1 \ \forall \boldsymbol{y} \in P$ by using the definition of the convex hull
- this proposition is deduced from the two previous ones and Farkas lemma, the proof is developed in the appendix

□

5.5 Proof of Farkas lemma

The homogeneous Farkas lemma is :

Theorem 5.12 (homogeneous Farkas lemma)**.** $\boldsymbol{a_1}^T\boldsymbol{x} \geq 0, ..., \boldsymbol{a_n}^T\boldsymbol{x} \geq 0$ *implies* $\boldsymbol{c}^T\boldsymbol{x} \geq 0$ *if and only if there exists* $(\lambda_i)_n$ *reals positives as:*

$$\sum_{i=1}^{n} \lambda_i \boldsymbol{a_i} = \boldsymbol{c}$$

Proof. The first direction is trivial.
For the second, if $\boldsymbol{c} \notin Cone(A)$, then $\boldsymbol{c}^T\boldsymbol{x} < 0$ is not possible. Let $\boldsymbol{d} \in \mathbb{R}^d$ with :

$$\boldsymbol{d}^T\boldsymbol{x} > \delta \ \forall \boldsymbol{x} \in Cone(A)$$

$$\boldsymbol{d}^T\boldsymbol{c} < \delta$$

Through the hyperplan of separation, there is $\delta < 0$ because $\mathbf{0} \in Cone(A)$. Taking for example $\frac{1}{\epsilon}\boldsymbol{a}_1 \in Cone(A)$, then $\boldsymbol{d}^T\boldsymbol{a}_1 > -\epsilon \ \forall \epsilon > 0$, and therefore $\boldsymbol{d}^T\boldsymbol{a}_1 \geq 0$, which gives $\boldsymbol{c}^T\boldsymbol{x} < 0$ impossible.
□

Theorem 5.13 (non homogeneous Farkas lemma)**.** *In* $\mathbb{R}^d$*, a system of inequalities,*
$\boldsymbol{a_1}^T\boldsymbol{x} \geq b_1, ..., \boldsymbol{a_n}^T\boldsymbol{x} \geq b_n$ *does not have a solution if and only if there is are*$(\lambda_i)_n$ *reals positives as:*

$$\sum_{i=1}^{n} \lambda_i \boldsymbol{a_i} = \mathbf{0}$$

$$\sum_{i=1}^{n} \lambda_i b_i = 1$$

Proof. The first way is self evident. The second comes from the Fourier Motzkin elimination process (detailed in the appendix), en normalizing the λ coming from the process.

□

To those theorems is preferred a more elegant version and a proof using the idea of polar duality :

Theorem 5.14 (Farkas lemma)**.** *Let* $g, f_1, ..., f_n$ *linear functions from* $\mathbb{R}^d$ *to* $\mathbb{R}$*, there is an equivalence between :*

$$g \in Cone(f_1, ..., f_n)$$

$$\bigcap_{i=1}^{n} \{f_i(\boldsymbol{x}) \geq 0\} \subset \{g(\boldsymbol{x}) \geq 0\}$$

Another interesting formulation used in [10] is worth mentioning, writing -1 the function that outputs -1:

Theorem 5.15 (Farkas lemma)**.** *Let $f_1, ..., f_n$ linear functions from $\mathbb{R}^d$ to $\mathbb{R}$, there is an equivalence :*

$$-1 \in Cone(f_1, ..., f_n)$$

$$\bigcap_{i=1}^{n} \{f_i(\boldsymbol{x}) \geq 0\} = \emptyset$$

Proof. The direct way is straightforward.

For the indirect way, let $f_i(\boldsymbol{x}) = \boldsymbol{y_i}^T \boldsymbol{x} + a_i$ and $\boldsymbol{z_i} = (\boldsymbol{y_i}, a_i)$ in $\mathbb{R}^{d+1}$.

Writing $A^* = \left\{\boldsymbol{x} | \boldsymbol{x}^T \boldsymbol{y} \geq 0 \ \forall \boldsymbol{y} \in A\right\}$ It is the dual for the cones.

Verifying that $\{\boldsymbol{z_1}, ..., \boldsymbol{z_n}\}^* \subset \boldsymbol{e_{n+1}}^*$.

Therefore $\boldsymbol{e_{n+1}} \in Cone(\boldsymbol{z_1}, ..., \boldsymbol{z_n})$ and then there is $\lambda_i \geq 0$ tels que $\sum_i \lambda_i f_i = 1$.

Easily follows $-1 \in Cone(f_1, ..., f_n)$ from the hypothesis.

□

A possible application : another proof of Helly's theorem using Farkas' lemma and Caratheodory s theorem. More details can be found in the appendix.

6 Number of linear regions in neural networks

Are now presented some results from the tropical approach applied to neural networks [16] and a result from polyhedron theory [13] regarding the number of linear regions in a neural network.

6.1 Tropical approach

The tropical approach makes it possible to apply theorems of combinatorics in order to better understand neural networks. The study of polytopes allows to obtain a result on the sum of Minkowski [7],

Theorem 6.1 (Gritzmann Sturmfels)**.** *Let $P_1, ...P_k$ be polytopes in $\mathbb{R}^d$, let m be the number of edges of $\{P_1, ...P_k\}$ not parallel, then the number of vertex in the Minkowski sum of those polytopes : $\bigoplus_{i=1}^{k} P_i$ is less than $2\sum_{j=0}^{d-1} \binom{m-1}{j}$.*

This theorem can be applied to zonotopes (which are Minkowski sums of line segments).

Corollary 6.1.1. *Let $P \in \mathbb{R}^{d+1}$ a zonotope from the sum of m line segments $P_1, ..., P_m$, let $\pi : \mathbb{R}^{d+1} \to \mathbb{R}^d$ deleting the last coordinate then in the general case, P has $\sum_{j=0}^{d} \binom{m}{j}$ vertices in its convex hull. If the line segments are parallel or some vertices are trivial,this value becomes an upper bound.*

Proof. Let V_1 and V_2 be the set vertices of the lower and upper convex envelopes of P, next to the projection $\pi : \mathbb{R}^{d+1} \to \mathbb{R}^d$ which removes the last component. According to the previous theorem 6.1 , P has n_1 vertices:

$$n_1 = 2\sum_{j=0}^{d} \binom{m-1}{j}$$

By property of the zonotopes we know that V_1 and V_2 have same cardinal and by definition of the convex hull $|V_1 \cup V_2| = n_1$.

Considering P' the projection by π of P into $\mathbb{R}^d$ which removes the last component. P' is a zonotope of dimension d because is generated by m line segments not 2 per 2 parallel. By applying the previous theorem P' to n_2 vertices :

$$n_2 = 2\sum_{j=0}^{d-1} \binom{m-1}{j}$$

For any v vertex of P, $\pi(v)$ is a vertex of P' if and only if v belongs to the upper convex hull and the lower convex hull : $|V_1 \cap V_2| = n_2$.

Therefore : $|V_1| = \frac{n_1 - n_2}{2} = \sum_{j=0}^{d} \binom{m}{j}$.

□

This result can be applied to neural networks:

Theorem 6.2. *Let $\nu : \mathbb{R}^d \to \mathbb{R}$ a neural network satisfying the previous conditions $(a) - (c)$, and if $n_l \geq d \; \forall l$ in $[1, L]$, then ν has at most $\prod_{l=1}^{L-1} \sum_{i=0}^{d} \binom{n_l}{i}$ linear regions. And for n real as $n \geq n_l \; \forall l$ then the number of linear regions of ν is bounded in $O(n^{d(L-1)})$.*

Proof. If $L = 2$: it is the previous corollary 6.1.1.

For $L \geq 3$ the proof is longer and is available in Zhang et Al. [16].

□

In the literature many methods exist to find higher bounds, such as sampling and using combinatorics.

6.2 Combinatorics approach

An interesting method for counting linear regions is based on an 'optimization' formulation called MILP: mixed integer linear programming, which is presented in Serra et al. [13]

The combinatorial idea is for a l layer composed of n_l neurons to create a vector $z_i^{(l)}$, i ranging from 1 à n_l. For a neuron i from layer l, $z_i^{(l)}$ is worth 0 or 1 upon the output of Relu (0 or not). Let ν^l the output of l.

Realizing that each different activation scheme corresponds to a linear region, possibly different from the previous ones, it is possible to deduce a 'classical' upper bound present in many works, by limiting by 2^{n_l} the number of linear regions created by a layer of n_l neurons [12].

Keeping the l notation for the layers (ranging from 1 to L), i for the neurons (ranging from 1 to n_l); at the $l$ieth layer, the network can be described by the system of equations:

1. $A_i^l p^{l-1} + b_i^l = p_i^l$
2. $p_i^l = h_i^l - \overline{h}_i^l$
3. $h_i^l \leq M z_i^l$
4. $\overline{h}_i^l \leq M(1 - z_i^l)$
5. $h_i^l \geq 0$
6. $\overline{h}_i^l \geq 0$
7. $z_i^l \in \{0, 1\}$

The insertion of the positive constant M is a fictional constraint imposed on the network in order to limit the space of the solutions.

Theorem 6.3. *If $|A_i^l \nu^{l-1} + b_i^l| \leq M$ for any acceptable value of ν^{l-1}, the formulation under constraints $(1) - (6)$ transforms $\boldsymbol{x} \in \mathbb{R}^{n_1}$ in $\nu(\boldsymbol{x}) \in \mathbb{R}^{n_L}$*

Proof. The proof by recurrence relies on network structure, the system of equations can be transposed. For a i neuron of the layer l :

If $A_i^l \nu^{l-1} + b_i^l > 0$, $(1) and (2)$ give $h_i^l - \overline{h}_i^l > 0$, (5), (6) and (7) implies $z_i^l = 1$ and then $A_i^l p^{l-1} + b_i^l = p_i^l$, therefore by recurrence $p_i^l = \nu_i^l$.

The same reasoning is used in cases where $A_i^l \nu^{l-1} + b_i^l < 0$ et $A_i^l \nu^{l-1} + b_i^l = 0$ □

To count the number of linear regions, Serra et al. [14] counts the solutions using a 'one-tree' method presented in Danna et al. [4].

These two methods show how the study of polytopes and linear programming make it possible to better understand neural networks.

7 Exploration, study of an application

7.1 Problem description

The 2 dimensions classification problem is defined as: class 0, in purple on fig. 10 corresponds to a donut (surface between 2 ellipses), class 1 in yellow represents the rest of the space. This figure is obtained by randomly sampling points in a grid and coloring each point according to its position.

The goal is to train neural networks of different architectures (number and width of layers) and to study the decision boundaries as well as the tropical varieties associated with these networks.

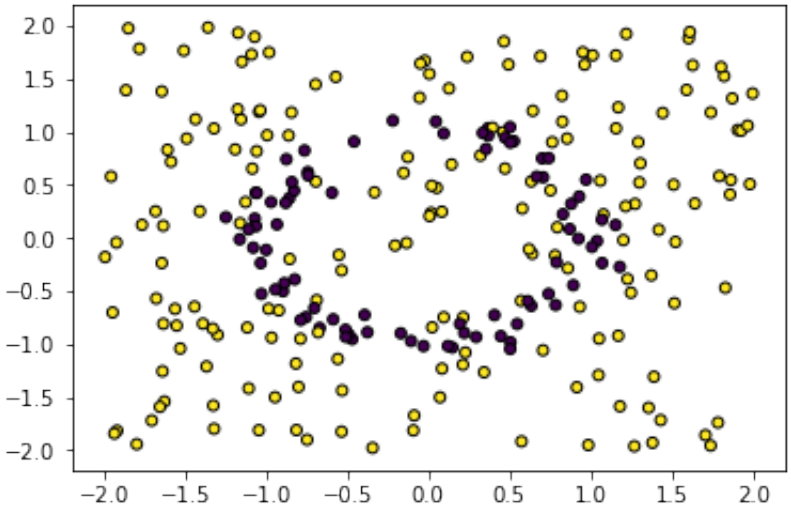

Figure 10: Example of classification problem in donut 2d

7.2 neural networks

Comparing three types of architecture :

1. networks with Relu activations composed of one hidden layer of width h (latent space dimension) $\mathbb{R}^2 \times \mathbb{R}^h \times \mathbb{R}$.
2. networks with Relu activations composed of two hidden layers of width h $\mathbb{R}^2 \times \mathbb{R}^h \times \mathbb{R}^h \times \mathbb{R}$.
3. a network with Maxout activations composed of one hidden layer. Maxout [6] is a type of activation who does 'max' between several linear combinations of the inputs.

For these 3 architectures, the inputs will be coordinates of the plane (dimension 2) and the output will be the class (0 or 1, in dimension 1)

The networks are trained and is measured the accuracy, between 0 and 1, which corresponds to the number of well classified points on the total number of points. fig. 11.

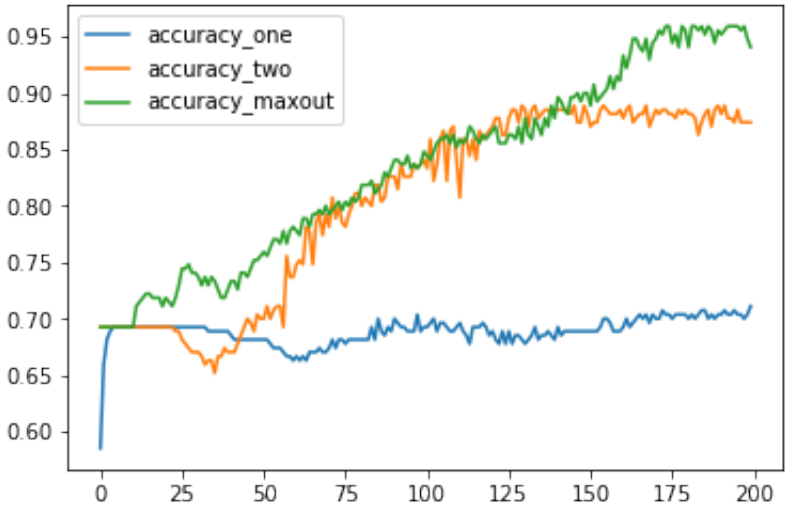

Figure 11: Networks accuracy during training

7.3 Decision frontiers, study of the tropical varieties

From the neural network, it is easy to display the following graphs :

1. the graph of the decision boundary, ie the line segments that separate the two classes. They are obtained simply by testing the network on a grid.
2. the points of discontinuities in the network: that is to say the changes of activation patterns, as described in Serra et al. [14]

7.4 Tropical varieties

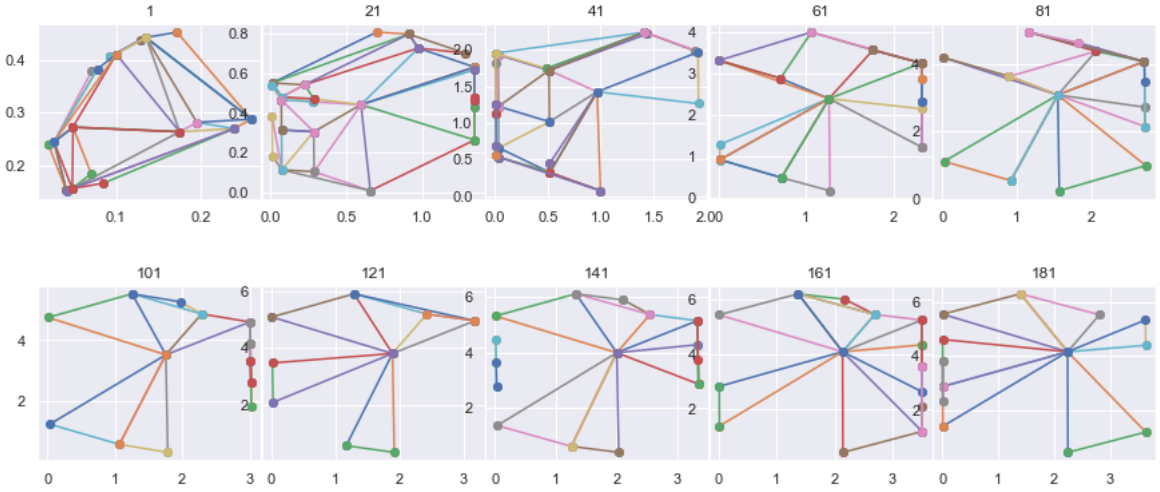

Figure 12: $\delta(f)$ for a network with 1 hidden layer, of width h = 10 at different steps of training

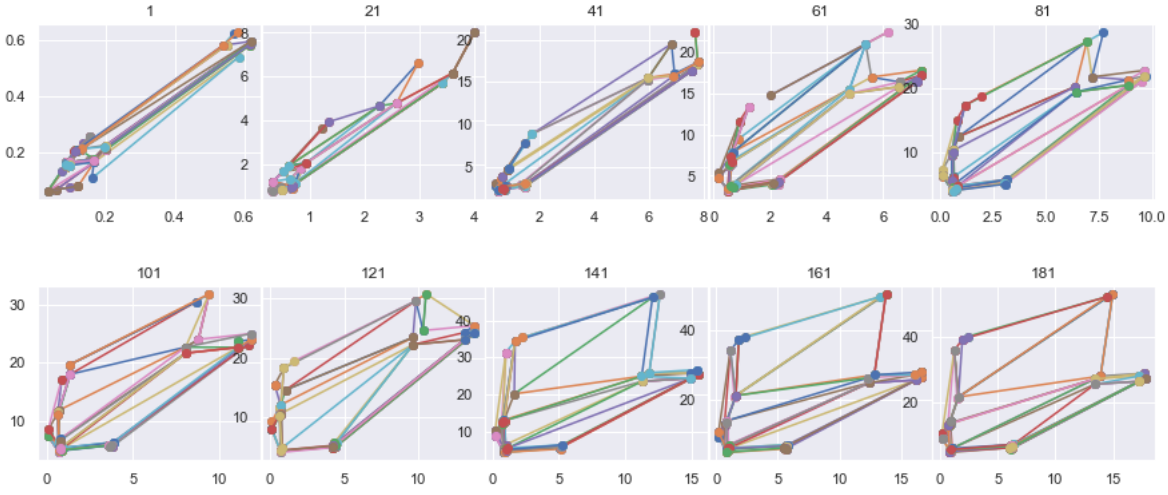

Figure 13: $\delta(f)$ for a network with 2 hidden layer, of width h = 4 at different steps of training

8 Conclusion

Thesis has shown that neural networks can be considered as tropical varieties with multiple following properties. Thus all the problems concerning the interpretation, the training and the depth and length of the architecture of a neural network can be studied in terms of hypersurfaces studies polytopes.

It has also shown some of the limits of the representation of computer neural networks, where rather simple architecture are difficult to process, even in the case of a two-dimensional input.

Some interesting possibilities for pursuing this approach include the interpretation of empirical properties of neural networks concerning kernel, regularization or depth of networks.

A Fourier Motzkin elimination

Consider the equations system in $\mathbb{R}^d$ under constraints :

$$i \in \{1, ..., n\}, \sum_{j=1}^{d} a_{ij} x_j \geq b_i$$

We are interested in the existence of a solution to this system. One of the most used methods is the Gaussian elimination method, and its generalization the Fourier Motzkin method. the principle consists in "eliminating" a variable if there are two coefficients of opposite signs in an inequality.

Considering x_1, writing $0 \leq k_1 \leq k2 \leq n$ the indexes of equations where x_1 is positive between 1 and k_1 and negative between k_1 and k_2.

Then for $i_1 \in \{1, ..., k_1\}$:

$$x_1 \geq \frac{b_{i_1}}{a_{i_1 1}} - \sum_{j=2}^{d} \frac{a_{i_1 j}}{a_{i_1 1}},\ a_{i_1 1} > 0$$

For $i_2 \in \{k_1 + 1, ..., k_2\}$:

$$x_1 \leq \frac{b_{i_2}}{a_{i_2 1}} - \sum_{j=2}^{d} \frac{a_{i_2 j}}{a_{i_2 1}},\ a_{i_2 1} < 0$$

By combining these two equations the problem becomes :

$$i_1 \in \{1, ..., k_1\}, i_2 \in \{k_1 + 1, ..., k_2\}, \sum_{j=2}^{d} (\frac{a_{i_1 j}}{a_{i_1 1}} - \frac{a_{i_2 j}}{a_{i_2 1}}) x_j \geq \frac{b_{i_1}}{a_{i_1 1}} - \frac{b_{i_2}}{a_{i_2 1}}$$

$$i \in \{k_2 + 1, ..., n\}, \sum_{j=1}^{d} a_{ij} x_j \geq b_i$$

Proposition A.1. *System A has a solution if and only if system B has a solution.*

Proof. The implication is given by the approach above.

Reciprocally, if $(x_2, ..., x_m)$ solution of B, choosing x_1 satisfying the condition above implies that $(x_1, x_2, ..., x_m)$ is a solution of B. In the case $k_2 = 0$, the theorem is trivial.

□

B Minkowski Weyl's theorem for cones

In addition to the Minkowski Weyl theorem for polytopes, the result in the general case is proven in[9] :

Definition B.1 (Polyhedral cone, conical hull)**.** Un cône polyèdral est une intersection de demi espaces vectoriels. $\forall \boldsymbol{x} \in Cone$, $\lambda \boldsymbol{x} \in Cone\ \forall \lambda \geq 0$.
The conical hull of a family of vectors is the set of linear combinations with positive coefficients of these vectors.

Proposition B.1 (Minkowski Weyl's theorem for cones)**.** *Every polyhedral cone is a conical hull and vice versa.*

C On Farkas's lemma

Farkas' lemma is the analog in complex analysis of the following lemma in linear algebra:

Proposition C.1. *Let $g, f_1, ..., f_n$ be affine formes in $\mathbb{R}^d$, there is an equivalence between :*

$$g \in Cone(f_1, ..., f_n)$$

$$\bigcap_{i=1}^{n} ker(f_i) \subset ker(g)$$

D Diagram of neural networks

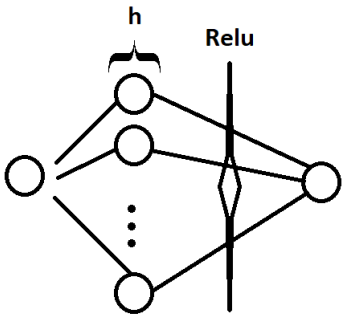

Figure 14: network with Relu activation composed of one hidden layer of width h

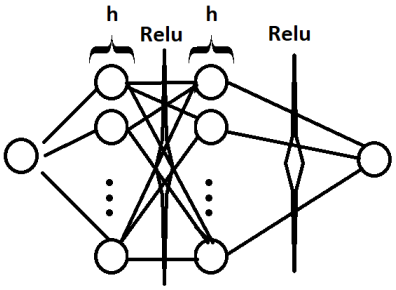

Figure 15: networks with Relu activations composed of two hidden layers of width h

E Code

The code is available on Git : https://github.com/ssggreg/Tropical-NN.

References

[1] C. Bradford Barber, David P. Dobkin, David P. Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Trans. Math. Softw.*, 22(4):469–483, December 1996.

[2] Mark de Berg, Otfried Cheong, Marc van Kreveld, and Mark Overmars. *Computational Geometry: Algorithms and Applications*. Springer-Verlag TELOS, Santa Clara, CA, USA, 3rd ed. edition, 2008.

[3] Vasileios Charisopoulos and Petros Maragos. A Tropical Approach to Neural Networks with Piecewise Linear Activations. *arXiv e-prints*, page arXiv:1805.08749, May 2018.

[4] Emilie Danna, Mary Fenelon, Zonghao Gu, and Roland Wunderling. Generating multiple solutions for mixed integer programming problems. In Matteo Fischetti and David P. Williamson, editors, *Integer Programming and Combinatorial Optimization*, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[5] Vincent Delos and Denis Teissandier. Minkowski Sum of Polytopes Defined by Their Vertices. *arXiv e-prints*, page arXiv:1412.2564, Dec 2014.

[6] Ian J. Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout Networks. *arXiv e-prints*, page arXiv:1302.4389, Feb 2013.

[7] Peter Gritzmann and Bernd Sturmfels. Minkowski addition of polytopes: Computational complexity and applications to gro¨bner bases. *SIAM J. Discret. Math.*, 6(2):246–269, May 1993.

[8] Michael Joswig. Essentials of tropical combinatorics. 2010.

[9] Francis Lazarus. cours de Géométrie Algorithmique. 2016.

[10] Joseph Lehec. Analyse convexe approfondie. 2014.

[11] Pavel Maur. Delaunay triangulation in 3 d. 2010.

[12] Guido Montúfar, Razvan Pascanu, Kyunghyun Cho, and Yoshua Bengio. On the Number of Linear Regions of Deep Neural Networks. *arXiv e-prints*, page arXiv:1402.1869, Feb 2014.

[13] Thiago Serra and Srikumar Ramalingam. Empirical Bounds on Linear Regions of Deep Rectifier Networks. *arXiv e-prints*, page arXiv:1810.03370, Oct 2018.

[14] Thiago Serra, Christian Tjandraatmadja, and Srikumar Ramalingam. Bounding and Counting Linear Regions of Deep Neural Networks. *arXiv e-prints*, page arXiv:1711.02114, Nov 2017.

[15] Jonathan Shewchuk. Cs 274 computational geometry. 2012.

[16] Liwen Zhang, Gregory Naitzat, and Lek-Heng Lim. Tropical geometry of deep neural networks. In *ICML*, 2018.