Interpretability of ReLU neural networks via truncation maps

Patrícia Muñoz Ewald

University of Texas at Austin

February 20, 2025

Outline

Includes joint work with Thomas Chen (UT Austin).

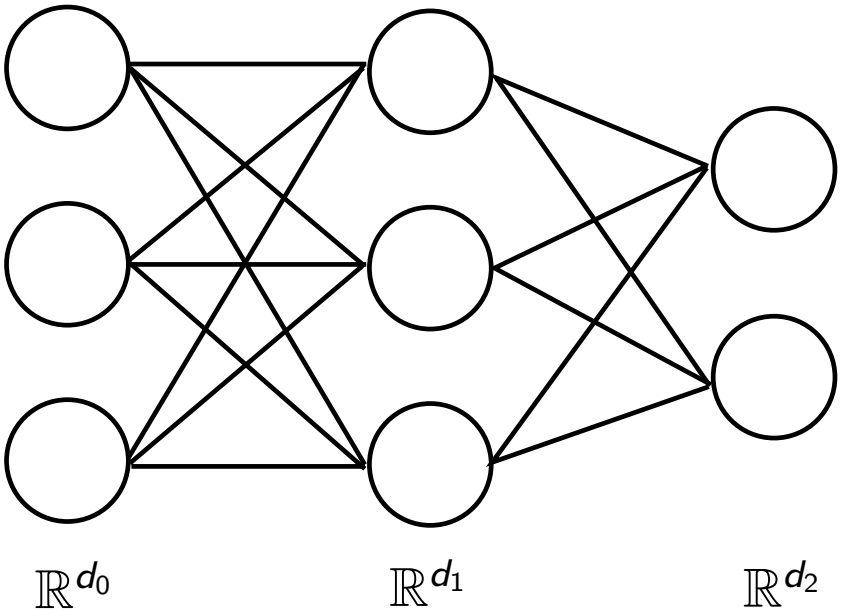
Main goal: To understand how a trained neural network works, and what the role of depth is.

1. Truncation map and cones
2. Multiclass classification of sequentially linearly separable data
3. Beyond sequentially linearly separable: Polyhedral cones

Consider a neural network

$$\begin{aligned} x^{(0)} &= x_0 \in \mathbb{R}^{d_0} \quad \text{initial input,} \\ x^{(\ell)} &= \sigma(W_\ell x^{(\ell-1)} + b_\ell) \in \mathbb{R}^{d_\ell}, \ \text{for } \ell = 1, \cdots, L-1, \\ x^{(L)} &= W_L x^{(L-1)} + b_L \in \mathbb{R}^{d_L}, \end{aligned} \tag{2.1}$$

with $\sigma(x)_i = \max\{0, x_i\}$.

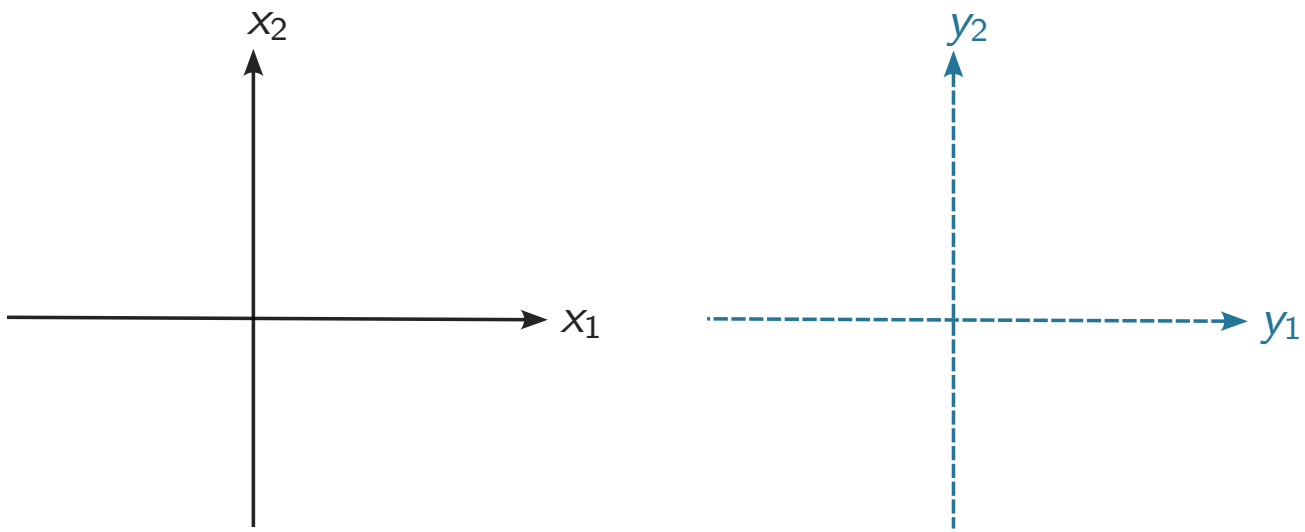


First goal: Change perspective.

Consider a neural network

$$
\begin{aligned}
x^{(0)} &= x_0 \in \mathbb{R}^{d_0} \quad \text{initial input,} \\
x^{(\ell)} &= \sigma(W_\ell x^{(\ell-1)} + b_\ell) \in \mathbb{R}^{d_\ell}, \ \text{for } \ell = 1, \cdots, L-1, \\
x^{(L)} &= W_L x^{(L-1)} + b_L \in \mathbb{R}^{d_L},
\end{aligned}
$$

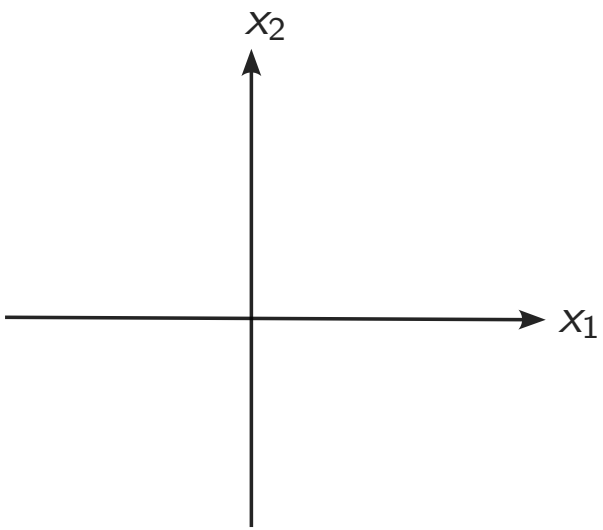
with $\sigma(x)_i = \max\{0, x_i\}$. Now observe

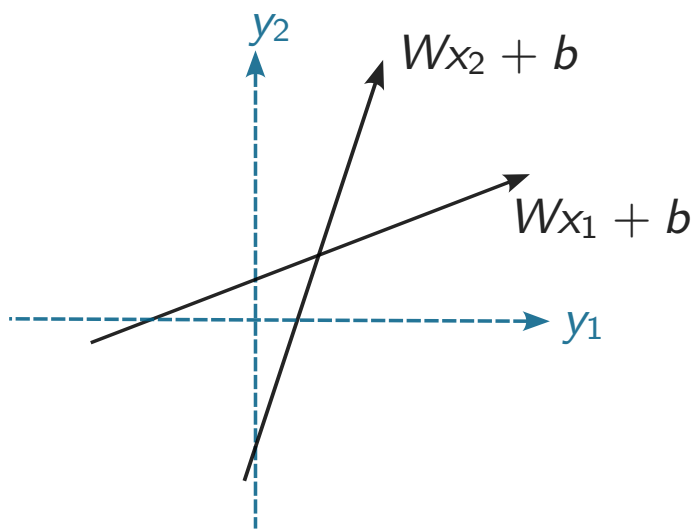


Consider a neural network

$$
\begin{aligned}
x^{(0)} &= x_0 \in \mathbb{R}^{d_0} \quad \text{initial input,} \\
x^{(\ell)} &= \sigma(W_\ell x^{(\ell-1)} + b_\ell) \in \mathbb{R}^{d_\ell}, \ \text{for } \ell = 1, \cdots, L-1, \\
x^{(L)} &= W_L x^{(L-1)} + b_L \in \mathbb{R}^{d_L},
\end{aligned}
$$

with $\sigma(x)_i = \max\{0, x_i\}$. Now observe

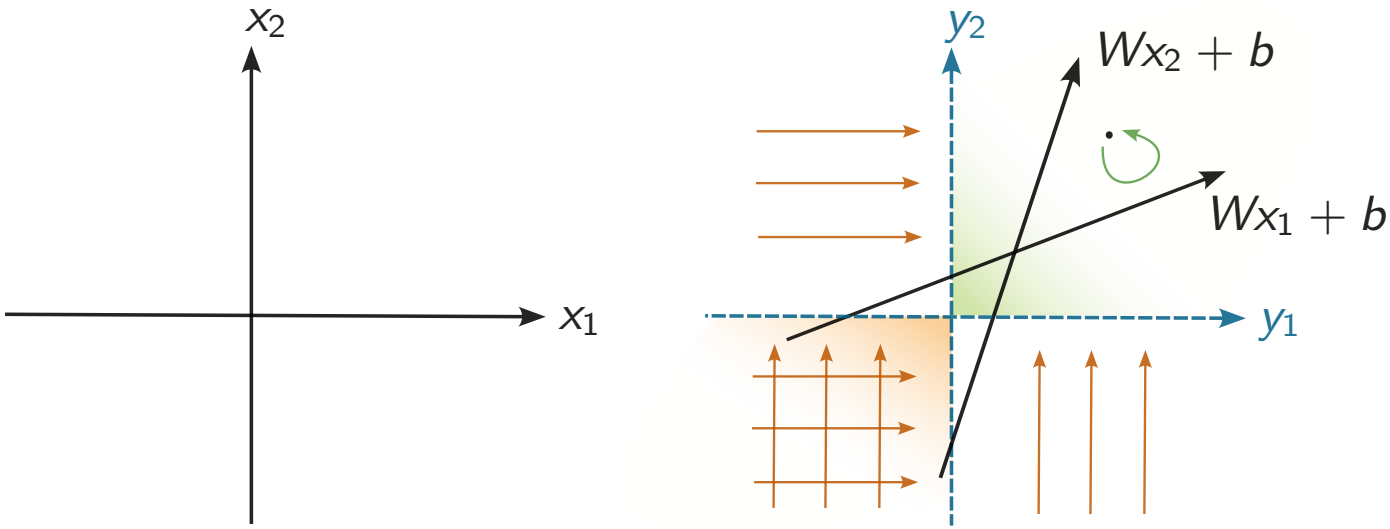




Consider a neural network

$$
\begin{aligned}
x^{(0)} &= x_0 \in \mathbb{R}^{d_0} \quad \text{initial input,} \\
x^{(\ell)} &= \sigma(W_\ell x^{(\ell-1)} + b_\ell) \in \mathbb{R}^{d_\ell}, \ \text{for } \ell = 1, \cdots, L-1, \\
x^{(L)} &= W_L x^{(L-1)} + b_L \in \mathbb{R}^{d_L},
\end{aligned}
$$

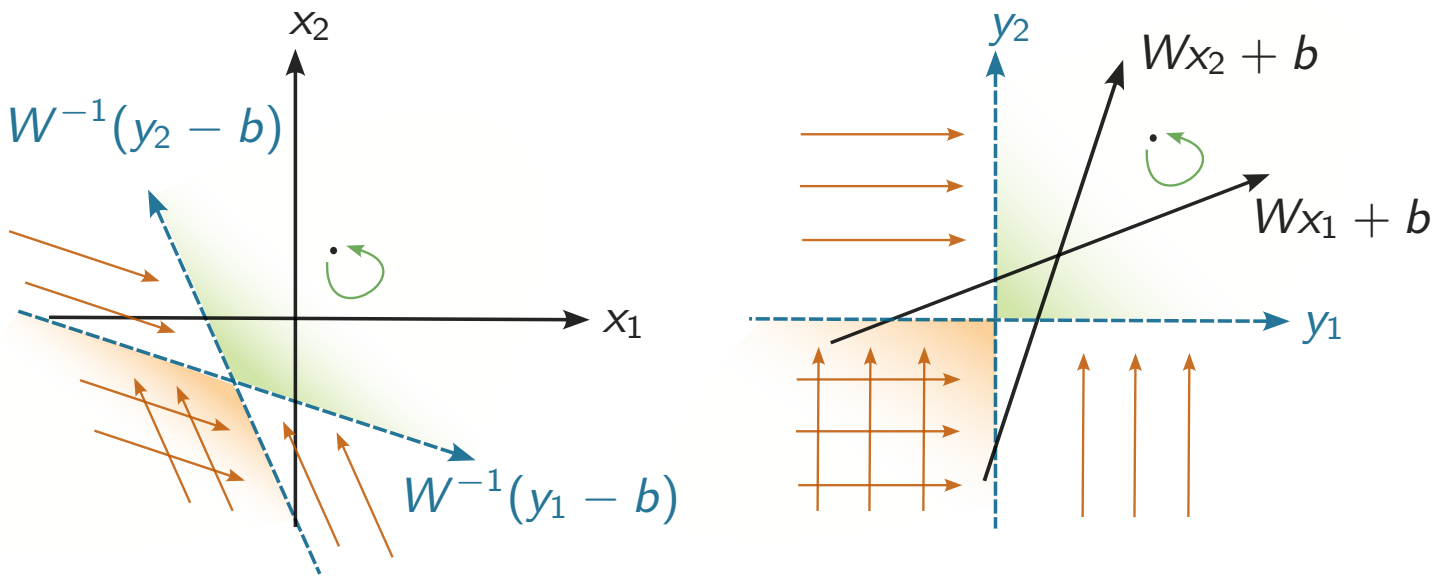
with $\sigma(x)_i = \max\{0, x_i\}$. Now observe



Consider a neural network

$$
\begin{aligned}
x^{(0)} &= x_0 \in \mathbb{R}^{d_0} \quad \text{initial input,} \\
x^{(\ell)} &= \sigma(W_\ell x^{(\ell-1)} + b_\ell) \in \mathbb{R}^{d_\ell}, \ \text{for } \ell = 1, \cdots, L-1, \\
x^{(L)} &= W_L x^{(L-1)} + b_L \in \mathbb{R}^{d_L},
\end{aligned}
$$

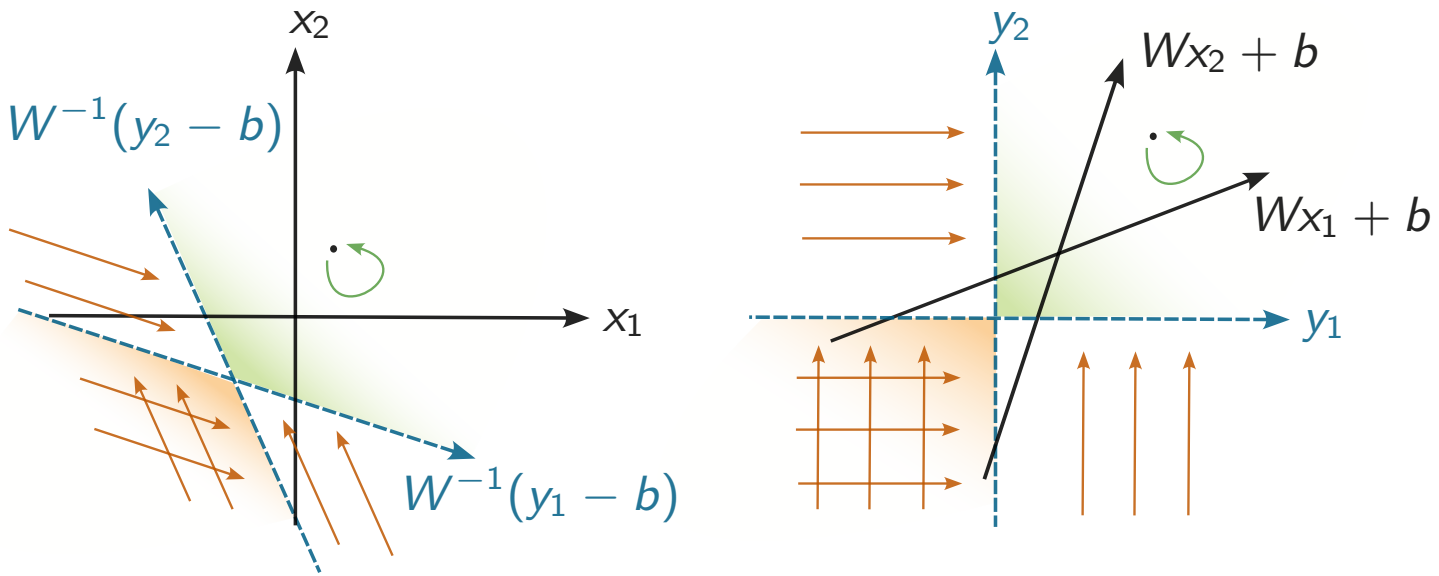
with $\sigma(x)_i = \max\{0, x_i\}$. Now observe



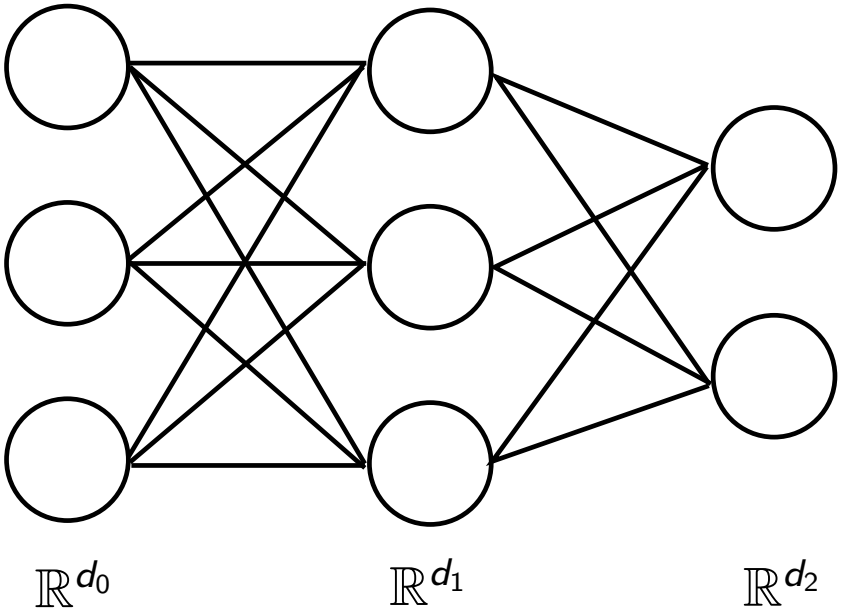
Definition (Chen–E. '23)

Given $W \in \mathbb{R}^{M_1 \times M_0}$ and $b \in \mathbb{R}^{M_1}$ we define the **truncation map**

$$\tau_{W,b} : \mathbb{R}^{M_0} \to \mathbb{R}^{M_0},$$
$$x \mapsto W^{-1}\left(\sigma(Wx + b) - b\right). \tag{2.2}$$



What about more layers?



Define the **cumulative parameters**

$$W^{(1)} := W_1, \quad W^{(\ell)} := W_\ell \cdots W_1 = W_\ell W^{(\ell-1)},$$
$$b^{(1)} := b_1, \quad b^{(\ell)} := W_\ell b^{(\ell-1)} + b_\ell, \tag{2.3}$$

and

$$x^{(\tau,0)} := x^{(0)}, \quad x^{(\tau,\ell)} := \tau_{W^{(\ell)},b^{(\ell)}}(x^{(\tau,\ell-1)}). \tag{2.4}$$

Recall $x^{(\ell)} = \sigma\left(W_\ell x^{(\ell-1)} + b_\ell\right)$ and $x^{(\tau,\ell)} = \tau_{W^{(\ell)},b^{(\ell)}}\left(x^{(\tau,\ell-1)}\right)$.

Proposition (Chen–E. '24)

If all weight matrices W_ℓ are surjective, $\ell = 1, \cdots, L$, then

$$x^{(\ell)} = W^{(\ell)} x^{(\tau,\,\ell)} + b^{(\ell)}, \tag{2.5}$$

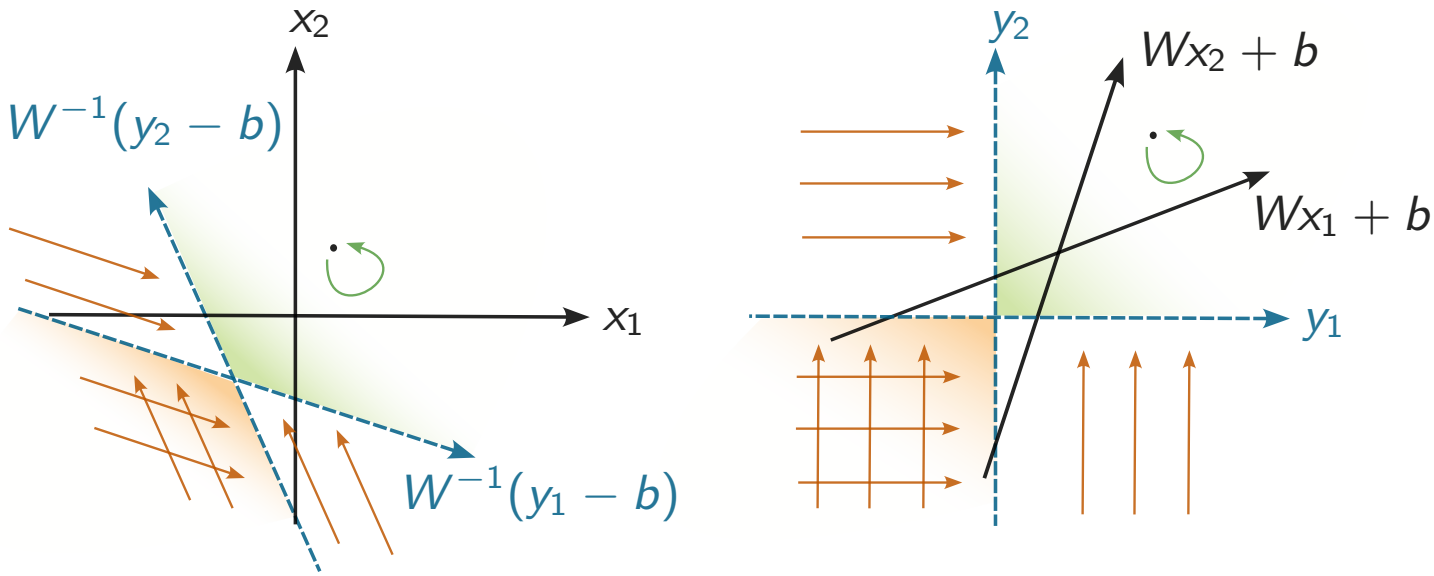
for $\ell = 1, \cdots, L-1$, and

$$x^{(L)} = W^{(L)} x^{(\tau,\,L-1)} + b^{(L)}. \tag{2.6}$$

A feedforward neural network with L layers can be seen as an application of $L-1$ endomorphisms τ on input space followed by an affine function in the last layer mapping to the output space.

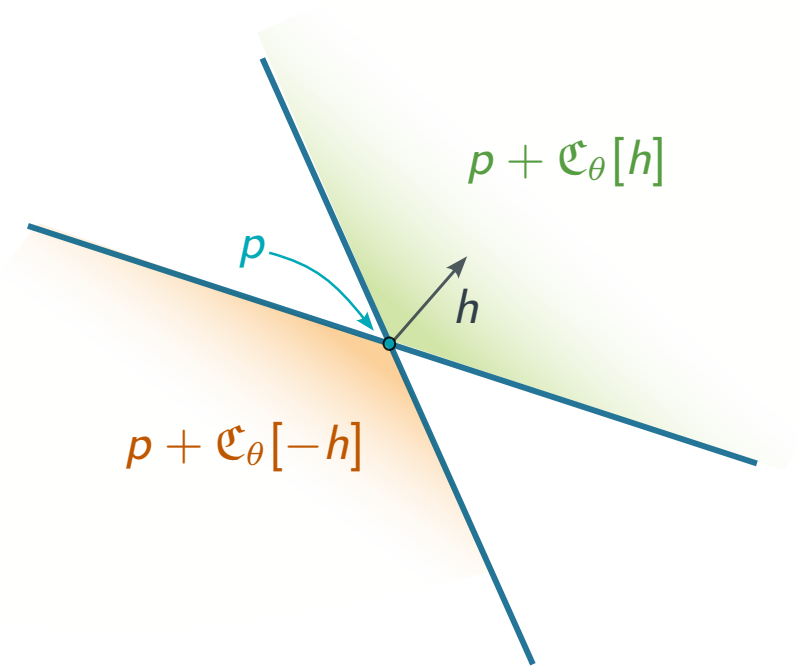
Truncation map and cones

Recall the picture



This completely describes the action of $\tau_{W,b}$ in 2 dimensions for invertible W. What about in general?

Consider the cone $p + \mathfrak{C}_\theta[h] \subseteq \mathbb{R}^n$ centered around a unit vector h and based at p:



Lemma (Chen–E. '24)

There are W and b such that

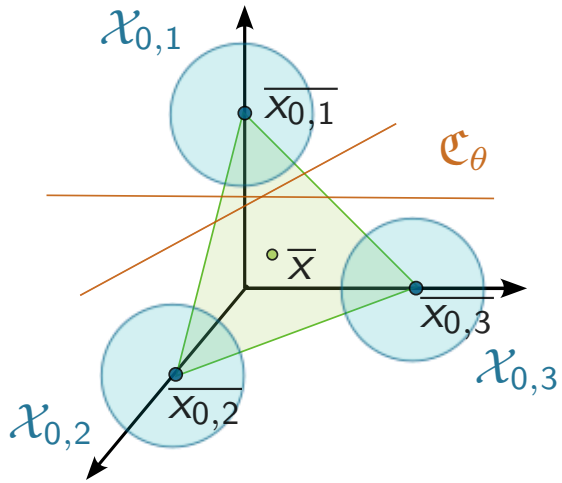
$$\tau_{W,b}(x) = \begin{cases} x, & x \in p + \mathfrak{C}_\theta[h], \\ p, & x \in p + \mathfrak{C}_\theta[-h]. \end{cases} \tag{2.7}$$

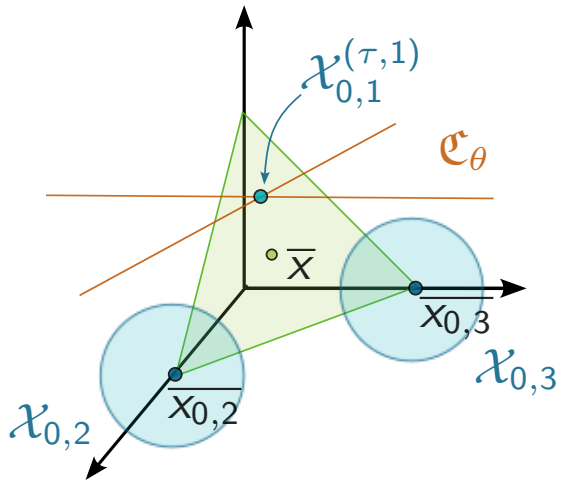
Theorem (Chen–E. '24)

Consider a set of training data $\mathcal{X}_0 = \bigcup_{j=1}^{Q} \mathcal{X}_{0,j} \subseteq \mathbb{R}^{d_0}$, *with* Q *classes associated to linearly independent labels* $y_j \in \mathbb{R}^Q$. *If the data is sufficiently clustered, then there exists a ReLU neural network with* $Q+1$ *layers which interpolates the data,*

$$\mathbb{R}^{d_0} \xrightarrow{\tau^{(1)}} \mathbb{R}^{d_0} \longrightarrow \cdots \xrightarrow{\tau^{(Q)}} \mathbb{R}^{d_0} \xrightarrow{W^{(Q+1)}} \mathbb{R}^Q. \tag{3.1}$$

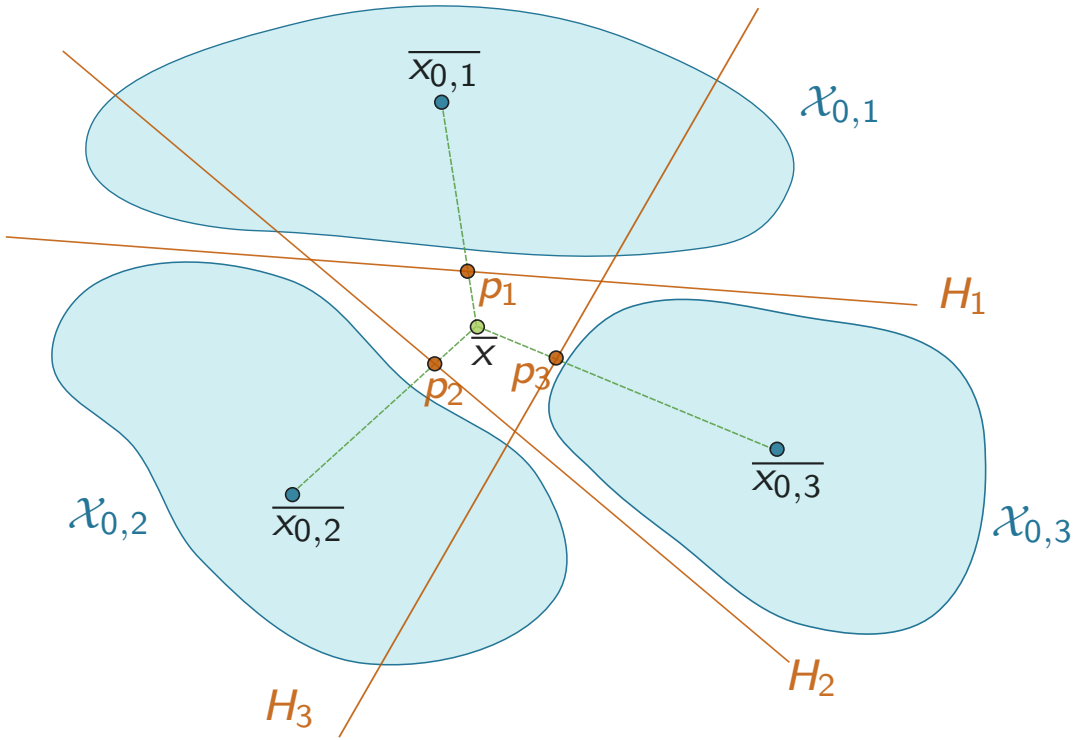
Proof sketch.





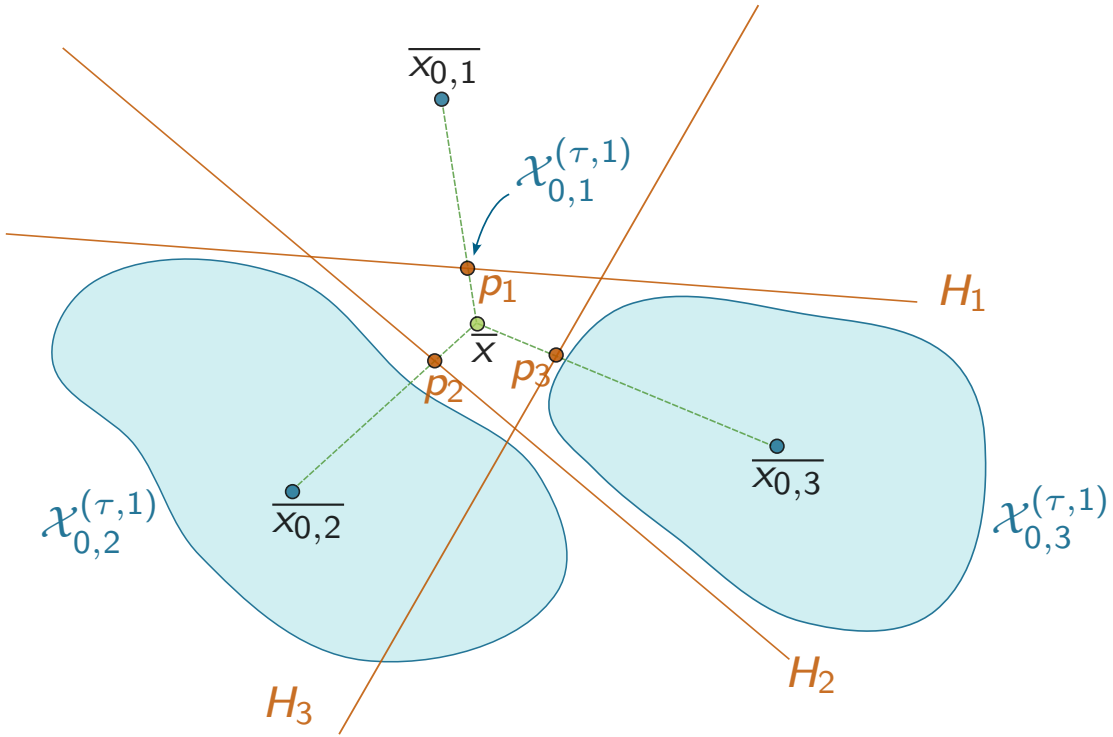
Definition (Chen–E. '24)

We say a dataset is **sequentially linearly separable** if there exists an ordering of the classes such that for each $j = 1, \cdots, Q$, there exists a point p_j in the line segment $\{\overline{x} + (1-t)\overline{x_{0,j}},\ t \in (0,1)\}$ and a hyperplane passing through p_j that separates $\mathcal{X}_{0,j}$ and $\bigcup_{i<j}\{p_i\} \cup \bigcup_{i>j} \mathcal{X}_{0,i}$.



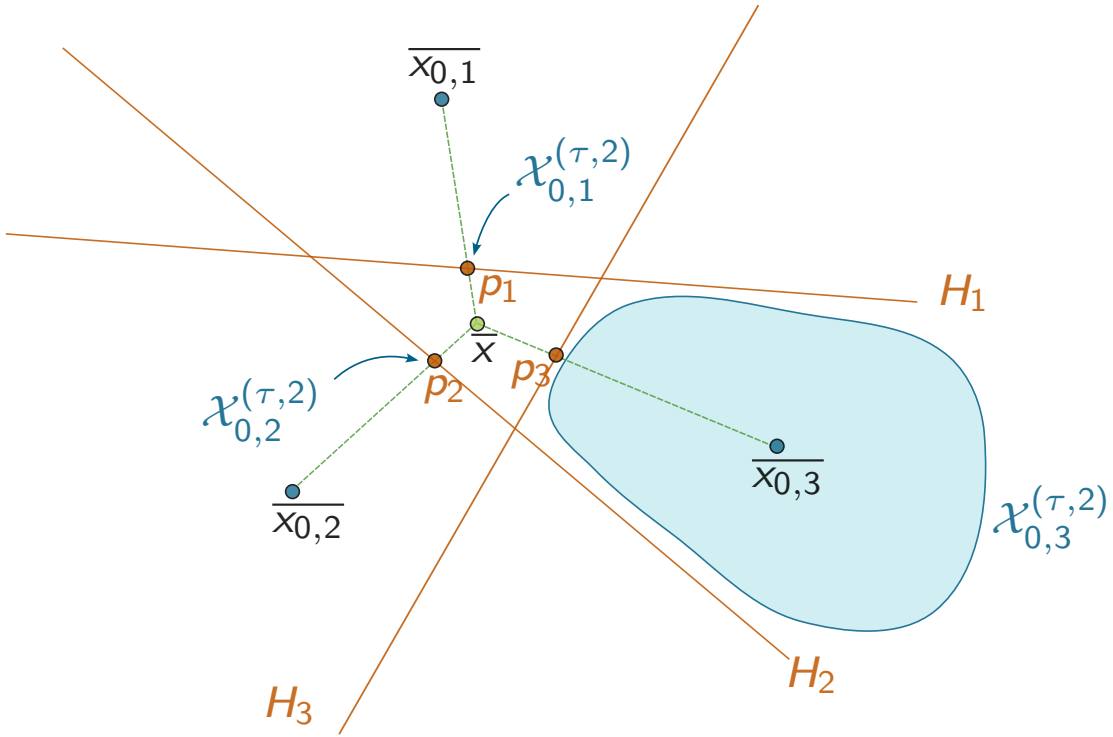
Definition (Chen–E. '24)

We say a dataset is **sequentially linearly separable** if there exists an ordering of the classes such that for each $j = 1, \cdots, Q$, there exists a point p_j in the line segment $\{\overline{x} + (1-t)\overline{x_{0,j}},\ t \in (0,1)\}$ and a hyperplane passing through p_j that separates $\mathcal{X}_{0,j}$ and $\bigcup_{i<j}\{p_i\} \cup \bigcup_{i>j} \mathcal{X}_{0,i}$.



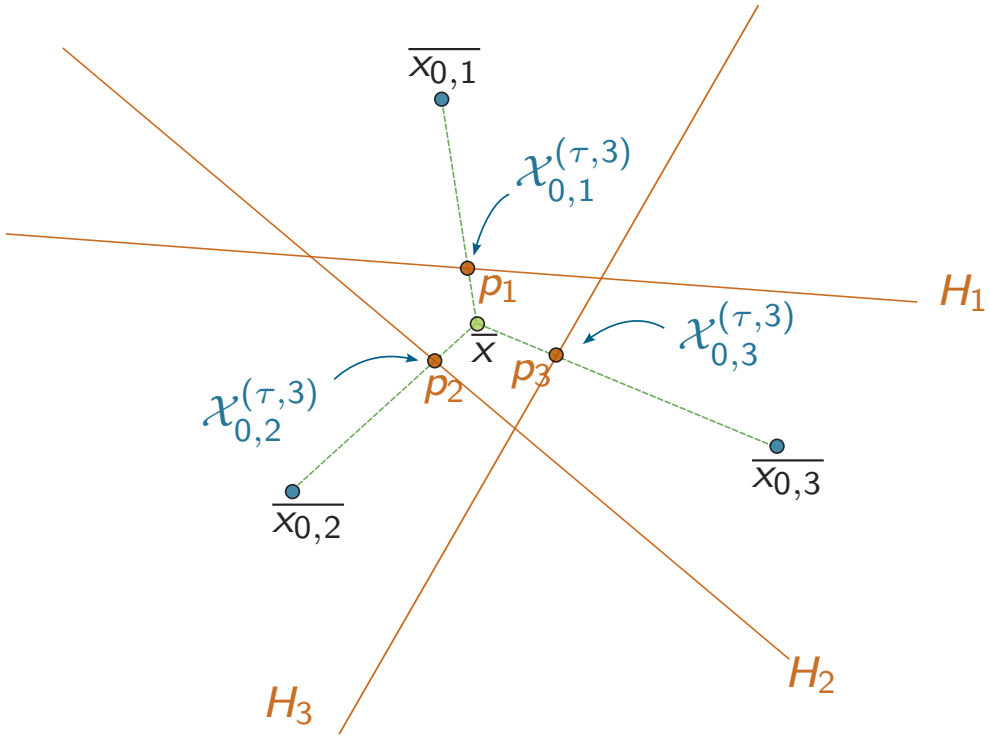
Definition (Chen–E. '24)

We say a dataset is **sequentially linearly separable** if there exists an ordering of the classes such that for each $j = 1, \cdots, Q$, there exists a point p_j in the line segment $\{\overline{x} + (1-t)\overline{x_{0,j}},\ t \in (0,1)\}$ and a hyperplane passing through p_j that separates $\mathcal{X}_{0,j}$ and $\bigcup_{i<j}\{p_i\} \cup \bigcup_{i>j} \mathcal{X}_{0,i}$.



Definition (Chen–E. '24)

We say a dataset is **sequentially linearly separable** if there exists an ordering of the classes such that for each $j = 1, \cdots, Q$, there exists a point p_j in the line segment $\{\overline{x} + (1-t)\overline{x_{0,j}},\ t \in (0,1)\}$ and a hyperplane passing through p_j that separates $\mathcal{X}_{0,j}$ and $\bigcup_{i<j}\{p_i\} \cup \bigcup_{i>j} \mathcal{X}_{0,i}$.

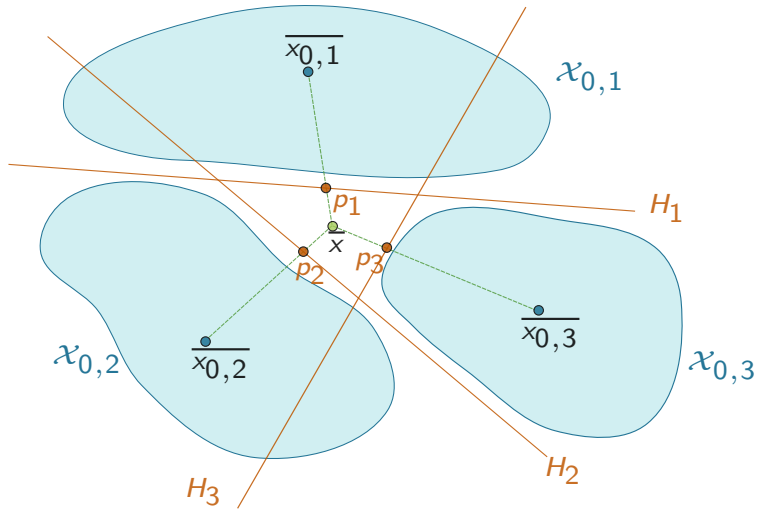


Theorem (Chen–E. '24)

If the data is sequentially linearly separable, then a ReLU neural network with $Q+1$ layers of size $d_0 = \cdots = d_Q = M \geq Q$, $d_{Q+1} = Q$, attains a degenerate global minimum with zero training cost, which can be parametrized by

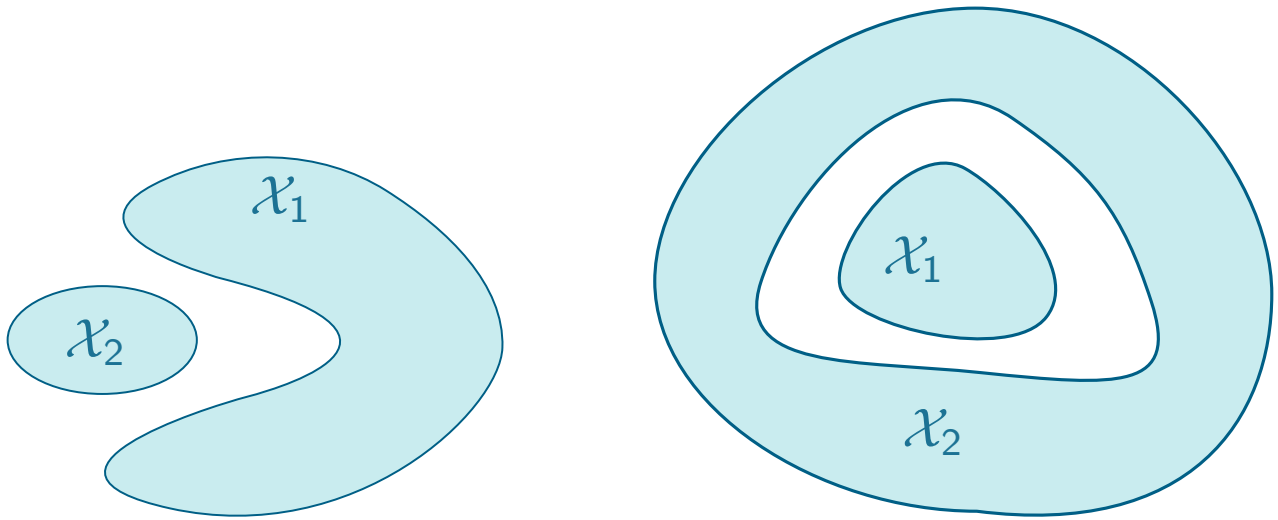
$$\{(\theta_\ell, \nu_\ell, \mu_\ell)\}_{\ell=1}^{Q} \subseteq (0,\pi) \times \mathbb{R}^M \times (0,1), \tag{3.2}$$

corresponding to triples of angles, normal vectors and a line segment of base points (resp.) describing cones and hyperplanes.



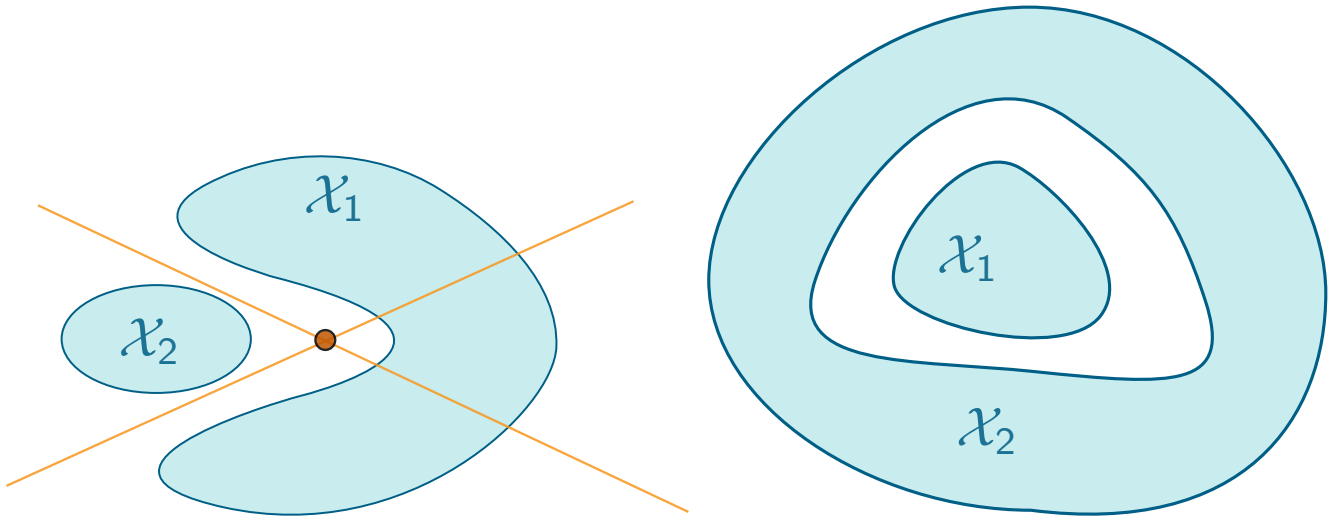
Beyond sequentially linearly separable

We would like to obtain similar results for data that is not sequentially linearly separable. For simplicity, consider binary classification.



Beyond sequentially linearly separable

We would like to obtain similar results for data that is not sequentially linearly separable. For simplicity, consider binary classification.



Polyhedral cones

Given $W \in GL(n)$ and $b \in \mathbb{R}^n$, we can find p and $(v_i)_{i=1}^m \in \mathbb{R}^n$ such that, if

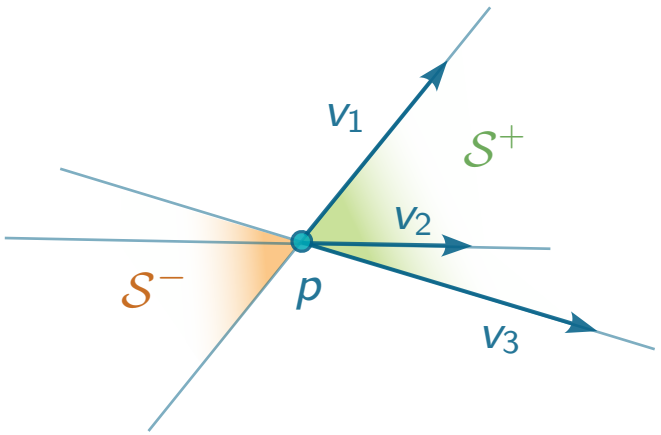
$$x = p + \sum_{i=1}^{m} a_i(x) v_i,$$

then

$$\tau_{W,b}(x) = p + \sum_{a_j(x)>0} a_j(x) v_j.$$

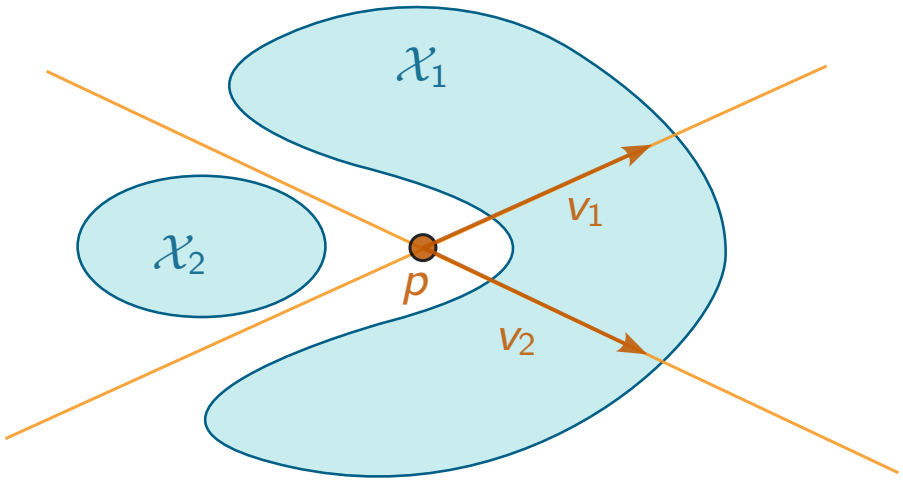
Define the negative sector

$$\mathcal{S}^- := \left\{ p + \sum_{i=1}^{n} a_i v_i : a_i \leq 0, i = 1, \cdots, n \right\}.$$



Proposition (E., *in preparation*)

Suppose $\mathcal{X} = \mathcal{X}_1 \cup \mathcal{X}_2 \subseteq \mathbb{R}^n$, and there exist p and $(v_i)_{i=1}^m \in \mathbb{R}^n$ such that $\mathcal{X}_1 \subseteq \mathcal{S}^-$ and $\mathcal{X}_2$ is a positive distance away from $\mathcal{S}^-$, for $n \geq m$. Then there exist $W : \mathbb{R}^n \to \mathbb{R}^m$ and $b \in \mathbb{R}^m$ such that the truncated data is linearly separable.



Topologically non-trivial data

What about concentric data? Take inspiration from support vector machines.

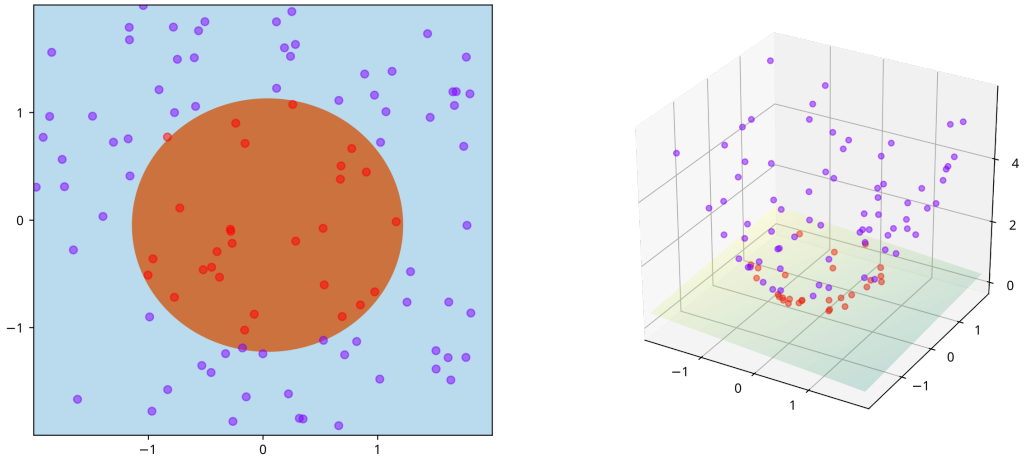


Figure: SVM with feature map $\phi(x) = (x, \|x\|^2)$.[1]

Must allow for dimension of layers to increase.

[1] Source: Wikipedia, created by user Shiyu Ji.

Proposition (E., *in preparation*)

Let $W : \mathbb{R}^n \to \mathbb{R}^m$, for $n < m$. There exists $\tilde{W} : \mathbb{R}^m \to \mathbb{R}^m$ such that $W = \tilde{W}\iota(n, m)$. If W is injective, then $\tilde{W}$ can be made invertible. In that case,

$$\sigma(Wx + b) = \tilde{W}\tau_{\tilde{W},b}(\tilde{x}) + b, \tag{4.1}$$

where $\tilde{x} = \iota(n, m)x \in \mathbb{R}^m$.

This means that it is possible to rewrite a neural network for which $d_0 < d_1$ and $d_1 \geq d_2 \geq \cdots \geq d_L$ in terms of truncation maps:

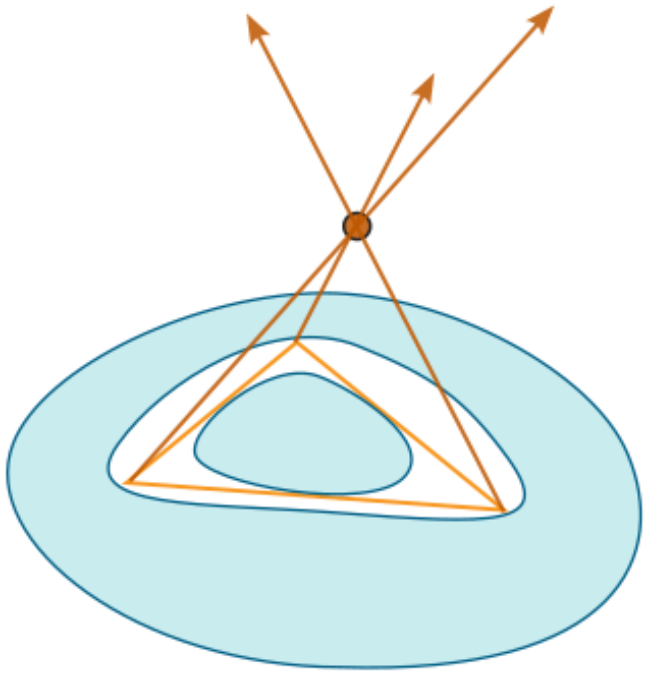
$$\mathbb{R}^{d_0} \xrightarrow{W_1,b_1} \mathbb{R}^{d_1} \xrightarrow{W_2,b_2} \mathbb{R}^{d_2} \longrightarrow \cdots$$

$$\mathbb{R}^{d_0} \xhookrightarrow{\iota} \mathbb{R}^{d_1} \xrightarrow{\tau_{\tilde{W}_1,b_1}} \mathbb{R}^{d_1} \xrightarrow{\tau^{(2)}} \mathbb{R}^{d_1} \longrightarrow \cdots$$

Theorem (E., *in preparation*)

Suppose $\mathcal{X} = \mathcal{X}_1 \cup \mathcal{X}_2 \subseteq \mathbb{R}^{d_0}$, and there exists a d_0-simplex separating $\mathcal{X}_1$ from $\mathcal{X}_2$. Then this data can be classified with zero loss by a ReLU neural network

$$\mathbb{R}^{d_0} \overset{\text{linear}}{\hookrightarrow} \mathbb{R}^{d_0+1} \xrightarrow{\tau^{(1)}} \mathbb{R}^{d_0+1} \xrightarrow{\tau^{(2)}} \mathbb{R}^{d_0+1} \xrightarrow{\text{linear}} \mathbb{R}^2.$$

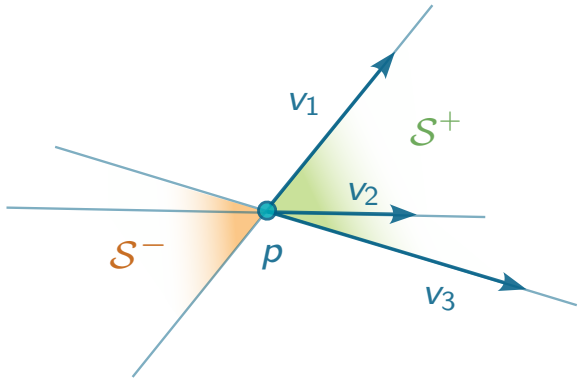


To summarize...

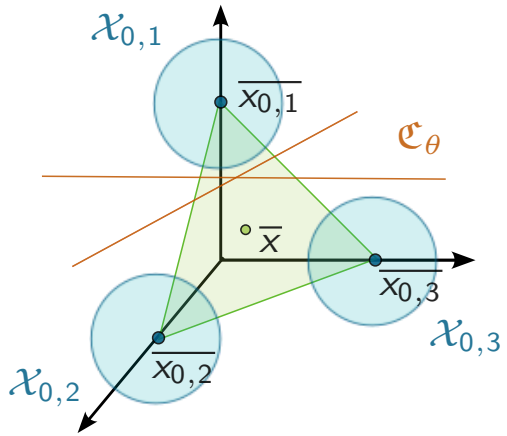
The truncation map allows us to reinterpret a deep neural network as a sequence of transformations on a fixed space,

$$x^{(L)} = W^{(L)}\, x^{(\tau,L-1)} + b^{(L)} = W^{(L)}_{LS}\, x^{(\tau,L-1)}.$$

For ReLU activation function, these transformations can be described in terms of (polyhedral) cones.



We can use this as a tool to construct interpolating classifiers.



Thank you!

Lemma (Chen–E. '24)

Consider the cone $p + \mathfrak{C}_\theta[h] \subseteq \mathbb{R}^n$ *and* $W : \mathbb{R}^n \to \mathbb{R}^m$, $b \in \mathbb{R}^m$ *such that*

$$W = \mathbf{1}_{m\times n} W_\theta R \text{ and } b = -Wp, \tag{6.1}$$

where

$$W_\theta = \tilde{R}\ \mathrm{diag}(1, \lambda(\theta, \theta_n), \cdots, \lambda(\theta, \theta_n))\, \tilde{R}^T, \tag{6.2}$$

with $\lambda(\theta, \theta_n) \leq 1$ *and* $R, \tilde{R} \in SO(n, \mathbb{R})$ *such that*

$$Rh = \tilde{R}e_1 = \left(\frac{1}{\sqrt{n}}, \cdots, \frac{1}{\sqrt{n}}\right)^T. \tag{6.3}$$

Then

$$\tau_{W,b}(x) = \begin{cases} (W^+W)\,x, & x \in p + \mathfrak{C}_\theta[h], \\ (W^+W)\,p, & x \in p + \mathfrak{C}_\theta[-h]. \end{cases} \tag{6.4}$$

Consider the cost

$$\mathcal{C}[(W_\ell, b_\ell)_{\ell=1}^L] := \sum_{j=1}^{Q} \frac{1}{N_j} \sum_{i=1}^{N_j} \left| x_{0,j,i}^{(L)} - y_j \right|^2 . \tag{6.5}$$

It can be decomposed into

$$\begin{aligned} \mathcal{C} &= \sum_{j=1}^{Q} \frac{1}{N_j} \sum_{i=1}^{N_j} \left| \Delta x_{0,j,i}^{(L)} \right|^2 + \sum_{j=1}^{Q} \left| \overline{x_{0,j}^{(L)}} - y_j \right|^2 \qquad (6.6) \\ &= \sum_{j=1}^{Q} \frac{1}{N_j} \sum_{i=1}^{N_j} \left| W^{(L)} \Delta x_{0,j,i}^{(\tau, L-1)} \right|^2 + \sum_{j=1}^{Q} \left| W^{(L)} \overline{x_{0,j}^{(\tau, L-1)}} + b^{(L)} - y_j \right|^2 . \end{aligned}$$

Then

$$\min_{(W_\ell, b_\ell)_{\ell=1}^L} \mathcal{C} \lesssim \min_{\underline{W}^{(L-1)}, \underline{b}^{(L-1)}} \sup_{x_0 \in \mathcal{X}_0} \left| Y \left(\overline{X_0^{red}}^{(\tau, L-1)} \right)^+ \Delta x_0^{(\tau, L-1)} \right|^2 \tag{6.7}$$

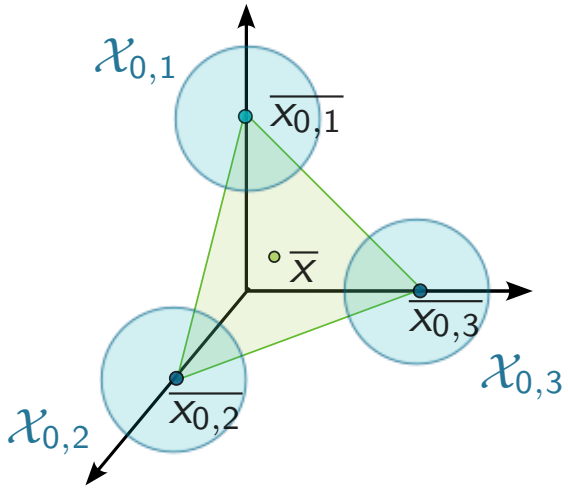
Since

$$\min_{(W_\ell, b_\ell)_{\ell=1}^{L}} \mathcal{C} \lesssim \min_{\underline{W}^{(L-1)}, \underline{b}^{(L-1)}} \sup_{x_0 \in \mathcal{X}_0} \left| Y \left(\overline{X_0^{red}}^{(\tau, L-1)} \right)^+ \Delta x_0^{(\tau, L-1)} \right|^2 , \tag{6.8}$$

we want $(\underline{W}^{(L-1)}, \underline{b}^{(L-1)})$ such that

$$\left(\overline{X_0^{red}}^{(\tau, L-1)} \right)^+ \Delta x_0^{(\tau, L-1)} \to 0 . \tag{6.9}$$

This is the expression for $\Delta x_0^{(\tau, L-1)}$ in "barycentric coordinates"



$$x = \sum_{j=1}^{Q} \kappa_j \overline{x_{0,j}}^{(\tau, L-1)} + \tilde{x}$$

where $\tilde{x} \in \left(\{ \overline{x_{0,j}}^{(\tau, L-1)} \}_{j=1}^{Q} \right)^{\perp}$.