Sparse factorizations of real polynomials & linear convolutional neural networks

Kathlén Kohn

KTH VETENSKAP OCH KONST

WASP WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

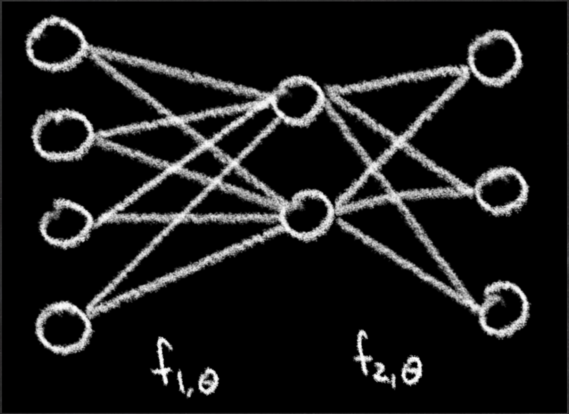
joint work with

Guido Montúfar
MPI MiS Leipzig & UCLA

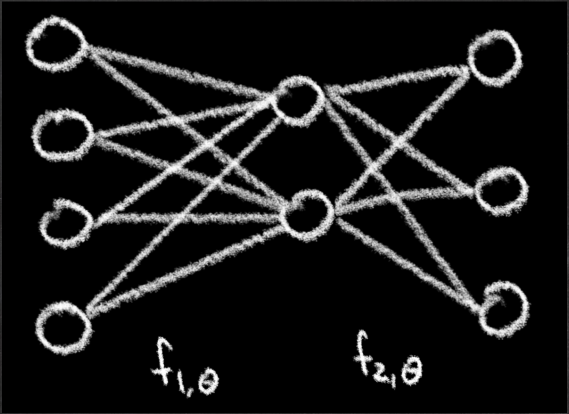
Vahid Shahverdi
KTH

Matthew Trager
Amazon Alexa AI, NYC

Neural networks



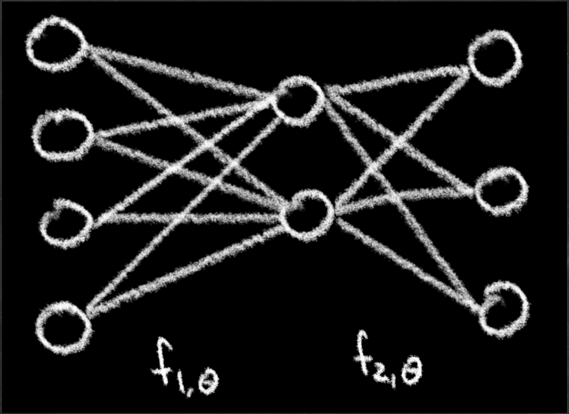
Neural networks



are parametrized families of functions

$$\mu : \mathbb{R}^N \longrightarrow \mathcal{M},$$
$$\theta \longmapsto f_{L,\theta} \circ \ldots \circ f_{1,\theta}$$

Neural networks



are parametrized families of functions

$$\mu : \mathbb{R}^N \longrightarrow \mathcal{M},$$
$$\theta \longmapsto f_{L,\theta} \circ \ldots \circ f_{1,\theta}$$

$\mathcal{M} =$ function space / neuromanifold, $L = \#$ layers

Training a network

Given training data $\mathcal{D}$, the goal is to minimize the loss

$$\mathbb{R}^N \xrightarrow{\mu} \mathcal{M} \xrightarrow{\ell_{\mathcal{D}}} \mathbb{R}.$$

Training a network

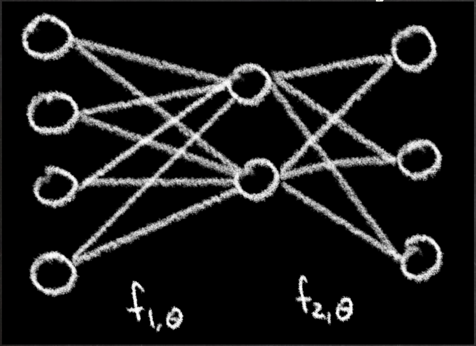
Given training data $\mathcal{D}$, the goal is to minimize the loss

$$\mathbb{R}^N \xrightarrow{\mu} \mathcal{M} \xrightarrow{\ell_{\mathcal{D}}} \mathbb{R}.$$

Geometric questions:

- How does the network architecture affect the geometry of the function space?
- How does the geometry of the function space impact the training of the network?

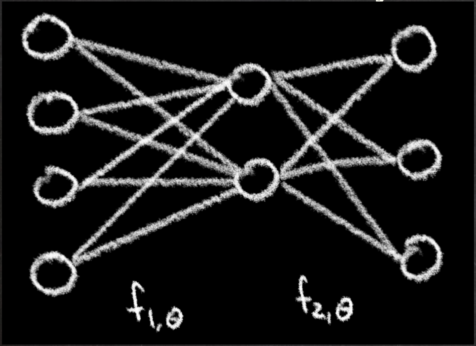
Linear fully-connected networks



In this example:

$$\mu : \mathbb{R}^{2\times 4} \times \mathbb{R}^{3\times 2} \longrightarrow \mathbb{R}^{3\times 4},$$
$$(W_1, W_2) \longmapsto W_2 W_1.$$

Linear fully-connected networks

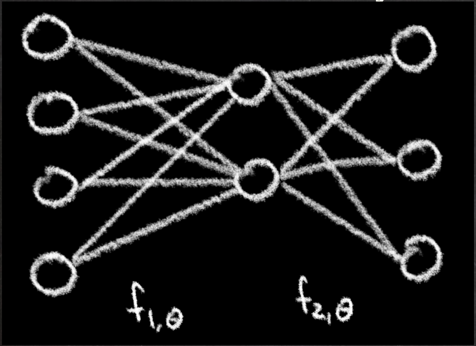


In this example:

$$\mu : \mathbb{R}^{2\times 4} \times \mathbb{R}^{3\times 2} \longrightarrow \mathbb{R}^{3\times 4}, \quad (W_1, W_2) \longmapsto W_2 W_1.$$

$$\mathcal{M} = \{W \in \mathbb{R}^{3\times 4} \mid \mathrm{rank}(W) \leq 2\}$$

Linear fully-connected networks



In this example:

$$\mu : \mathbb{R}^{2\times 4} \times \mathbb{R}^{3\times 2} \longrightarrow \mathbb{R}^{3\times 4},$$
$$(W_1, W_2) \longmapsto W_2 W_1.$$

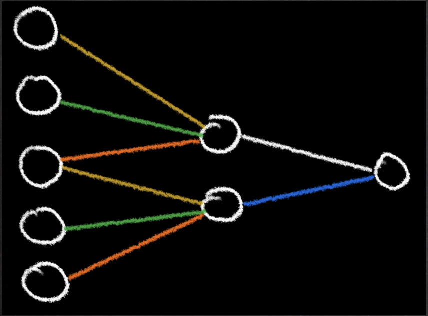
$$\mathcal{M} = \{W \in \mathbb{R}^{3\times 4} \mid \mathrm{rank}(W) \leq 2\}$$

In general:

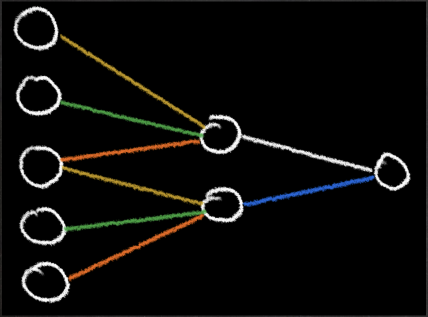
$$\mu : \mathbb{R}^{k_1\times k_0} \times \mathbb{R}^{k_2\times k_1} \times \ldots \times \mathbb{R}^{k_L\times k_{L-1}} \longrightarrow \mathbb{R}^{k_L\times k_0},$$
$$(W_1, W_2, \ldots, W_L) \longmapsto W_L \cdots W_2 W_1.$$

$\mathcal{M} = \{W \in \mathbb{R}^{k_L\times k_0} \mid \mathrm{rank}(W) \leq \min(k_0, \ldots, k_L)\}$ is a determinantal variety and we know its singularities etc.

Linear convolutional neural networks



Linear convolutional neural networks



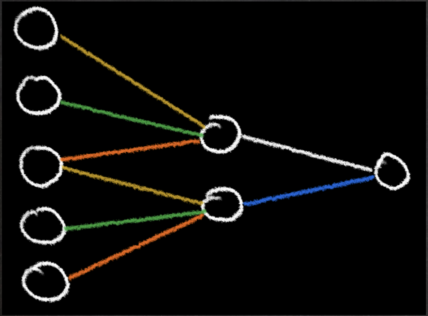
$$\mu : \mathbb{R}^3 \times \mathbb{R}^2 \longrightarrow \mathbb{R}^5,$$

$$(u, v) \longmapsto T_{v,1} T_{u,2}, \text{ where}$$

$$T_{u,2} = \begin{bmatrix} u_0 & u_1 & u_2 & 0 & 0 \\ 0 & 0 & u_0 & u_1 & u_2 \end{bmatrix}$$

$$T_{v,1} = \begin{bmatrix} v_2 & v_2 \end{bmatrix}$$

Linear convolutional neural networks



$$\mu : \mathbb{R}^3 \times \mathbb{R}^2 \longrightarrow \mathbb{R}^5, \quad (u, v) \longmapsto T_{v,1} T_{u,2}, \text{ where}$$

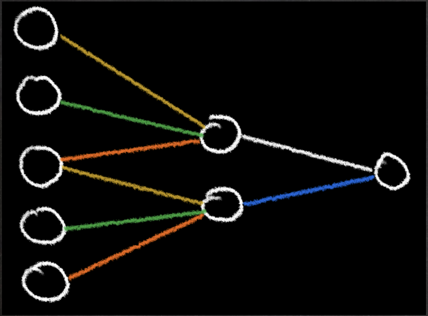
$$T_{u,2} = \begin{bmatrix} u_0 & u_1 & u_2 & 0 & 0 \\ 0 & 0 & u_0 & u_1 & u_2 \end{bmatrix}$$

$$T_{v,1} = \begin{bmatrix} v_2 & v_2 \end{bmatrix}$$

In general: $\mu : (w_1, \ldots, w_L) \mapsto T_{w_L,s_L} \cdots T_{w_1,s_1}$, where

$$T_{w,s} = \begin{bmatrix} w_0 & \cdots & w_s & \cdots & w_{k-1} & & & & \\ & & w_0 & & \cdots & & w_{k-1} & & \\ & & & \ddots & & & & \ddots & \\ & & & & & w_0 & & \cdots & w_{k-1} \end{bmatrix}$$

Linear convolutional neural networks



$$\mu : \mathbb{R}^3 \times \mathbb{R}^2 \longrightarrow \mathbb{R}^5,$$
$$(u, v) \longmapsto T_{v,1}\, T_{u,2}, \text{ where}$$

$$T_{u,2} = \begin{bmatrix} u_0 & u_1 & u_2 & 0 & 0 \\ 0 & 0 & u_0 & u_1 & u_2 \end{bmatrix}$$
$$T_{v,1} = \begin{bmatrix} v_2 & v_2 \end{bmatrix}$$

In general: $\mu : (w_1, \ldots, w_L) \mapsto T_{w_L, s_L} \cdots T_{w_1, s_1}$, where

$$T_{w,s} = \begin{bmatrix} w_0 & \cdots & w_s & \cdots & w_{k-1} & & & & \\ & & w_0 & & \cdots & & w_{k-1} & & \\ & & & \ddots & & & & \ddots & \\ & & & & & w_0 & & \cdots & w_{k-1} \end{bmatrix}$$

is a convolutional matrix of stride s with filter w

LCNs & sparse polynomial factorization

Observation: $\mu(w_1, \ldots, w_L) = T_{w_L, s_L} \cdots T_{w_1, s_1}$ is again a convolutional matrix of stride $s_1 \cdots s_L$.

LCNs & sparse polynomial factorization

Observation: $\mu(w_1, \ldots, w_L) = T_{w_L, s_L} \cdots T_{w_1, s_1}$ is again a convolutional matrix of stride $s_1 \cdots s_L$. Its filter can be computed via polynomial multiplication:

LCNs & sparse polynomial factorization

Observation: $\mu(w_1, \ldots, w_L) = T_{w_L,s_L} \cdots T_{w_1,s_1}$ is again a convolutional matrix of stride $s_1 \cdots s_L$. Its filter can be computed via polynomial multiplication:

For $S \in \mathbb{Z}_{>0}$, let

$$\pi_S : \mathbb{R}^k \longrightarrow \mathbb{R}[x^S, y^S]_{k-1},$$

$$w \longmapsto w_0 x^{S(k-1)} + w_1 x^{S(k-2)} y^S + \ldots + w_{k-2} x^S y^{S(k-2)} + w_{k-1} y^{S(k-1)}$$

LCNs & sparse polynomial factorization

Observation: $\mu(w_1, \ldots, w_L) = T_{w_L,s_L} \cdots T_{w_1,s_1}$ is again a convolutional matrix of stride $s_1 \cdots s_L$. Its filter can be computed via polynomial multiplication:

For $S \in \mathbb{Z}_{>0}$, let

$$\pi_S : \mathbb{R}^k \longrightarrow \mathbb{R}[x^S, y^S]_{k-1},$$
$$w \longmapsto w_0 x^{S(k-1)} + w_1 x^{S(k-2)} y^S + \ldots + w_{k-2} x^S y^{S(k-2)} + w_{k-1} y^{S(k-1)}$$

and $\pi_S(T_{w,s}) := \pi_S(w)$. Then:

LCNs & sparse polynomial factorization

Observation: $\mu(w_1, \ldots, w_L) = T_{w_L, s_L} \cdots T_{w_1, s_1}$ is again a convolutional matrix of stride $s_1 \cdots s_L$. Its filter can be computed via polynomial multiplication:

For $S \in \mathbb{Z}_{>0}$, let

$$\pi_S : \mathbb{R}^k \longrightarrow \mathbb{R}[x^S, y^S]_{k-1},$$
$$w \longmapsto w_0 x^{S(k-1)} + w_1 x^{S(k-2)} y^S + \ldots + w_{k-2} x^S y^{S(k-2)} + w_{k-1} y^{S(k-1)}$$

and $\pi_S(T_{w,s}) := \pi_S(w)$. Then:

$$\pi_1(\mu(w_1, \ldots, w_L)) = \pi_{S_L}(w_L) \cdots \pi_{S_1}(w_1), \text{ where } S_i := s_1 \cdots s_{i-1}.$$

LCNs & sparse polynomial factorization

Observation: $\mu(w_1, \ldots, w_L) = T_{w_L, s_L} \cdots T_{w_1, s_1}$ is again a convolutional matrix of stride $s_1 \cdots s_L$. Its filter can be computed via polynomial multiplication:

For $S \in \mathbb{Z}_{>0}$, let

$$\pi_S : \mathbb{R}^k \longrightarrow \mathbb{R}[x^S, y^S]_{k-1},$$
$$w \longmapsto w_0 x^{S(k-1)} + w_1 x^{S(k-2)} y^S + \ldots + w_{k-2} x^S y^{S(k-2)} + w_{k-1} y^{S(k-1)}$$

and $\pi_S(T_{w,s}) := \pi_S(w)$. Then:

$$\pi_1(\mu(w_1, \ldots, w_L)) = \pi_{S_L}(w_L) \cdots \pi_{S_1}(w_1), \text{ where } S_i := s_1 \cdots s_{i-1}.$$

Hence, we reinterpret μ as

$$\mu : \mathbb{R}[x^{S_1}, y^{S_1}]_{d_1} \times \ldots \times \mathbb{R}[x^{S_L}, y^{S_L}]_{d_L} \longrightarrow \mathbb{R}[x, y]_{d_1 S_1 + \ldots + d_L S_L},$$
$$(P_1, \ldots, P_L) \longmapsto P_L \cdots P_1$$

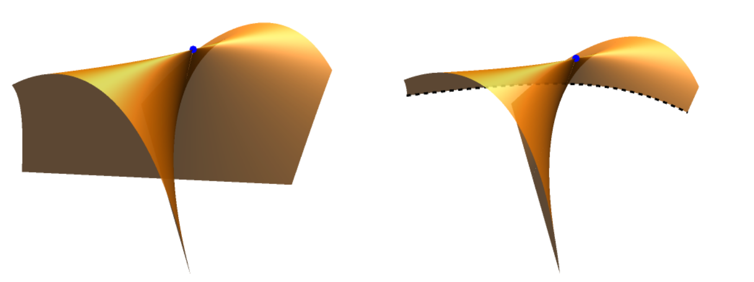
LCN function spaces

$$\mu : \mathbb{R}[x^{S_1}, y^{S_1}]_{d_1} \times \ldots \times \mathbb{R}[x^{S_L}, y^{S_L}]_{d_L} \longrightarrow \mathbb{R}[x, y]_d, \text{where } d := \sum_i d_i S_i$$
$$(P_1, \ldots, P_L) \longmapsto P_L \cdots P_1,$$

LCN function spaces

$$\mu : \mathbb{R}[x^{S_1}, y^{S_1}]_{d_1} \times \ldots \times \mathbb{R}[x^{S_L}, y^{S_L}]_{d_L} \longrightarrow \mathbb{R}[x,y]_d, \text{where } d := \sum_i d_i S_i$$
$$(P_1, \ldots, P_L) \longmapsto P_L \cdots P_1,$$

Theorem: The function space $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} = \mathrm{im}(\mu)$ is a semi-algebraic, Euclidean-closed subset of $\mathbb{R}[x,y]_d$ of dimension $d_1 + \ldots + d_L + 1$.

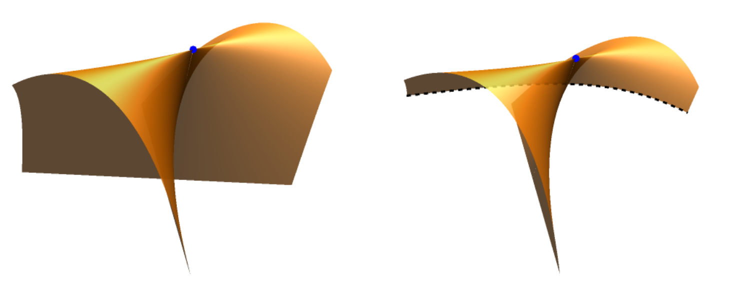


$\mu : \mathbb{R}[x,y]_2 \times \mathbb{R}[x^2,y^2]_1 \to \mathbb{R}[x,y]_4$ $\qquad$ $\mu : \mathbb{R}[x,y]_1 \times \mathbb{R}[x,y]_1 \times \mathbb{R}[x^2,y^2]_1 \to \mathbb{R}[x,y]_4$

Reducing an LCN architecture

$\mu : \mathbb{R}[x,y]_2 \times \mathbb{R}[x^2,y^2]_1 \to \mathbb{R}[x,y]_4$

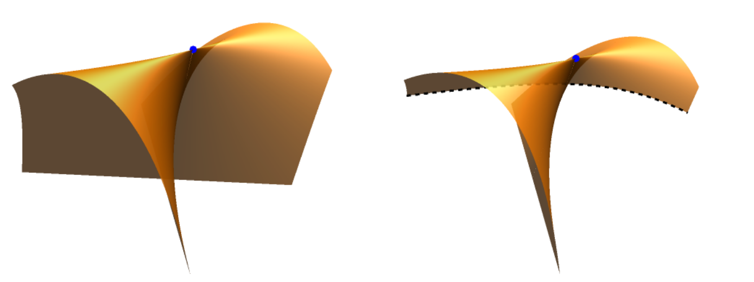
$\mu : \mathbb{R}[x,y]_1 \times \mathbb{R}[x,y]_1 \times \mathbb{R}[x^2,y^2]_1 \to \mathbb{R}[x,y]_4$



$$\begin{array}{ccccc} \mathbb{R}[x,y]_1 \times \mathbb{R}[x,y]_1 & \times & \mathbb{R}[x^2,y^2]_1 & \longrightarrow & \mathcal{M}_{(1,1,1),(1,1,2)} \\ \downarrow & & \downarrow & & \downarrow \\ \mathbb{R}[x,y]_2 & \times & \mathbb{R}[x^2,y^2]_1 & \longrightarrow & \mathcal{M}_{(2,1),(1,2)} \end{array}$$

Reducing an LCN architecture

$\mu : \mathbb{R}[x,y]_2 \times \mathbb{R}[x^2,y^2]_1 \to \mathbb{R}[x,y]_4$ $\qquad$ $\mu : \mathbb{R}[x,y]_1 \times \mathbb{R}[x,y]_1 \times \mathbb{R}[x^2,y^2]_1 \to \mathbb{R}[x,y]_4$



$$\begin{array}{ccccc} \mathbb{R}[x,y]_1 \times \mathbb{R}[x,y]_1 & \times & \mathbb{R}[x^2,y^2]_1 & \longrightarrow & \mathcal{M}_{(1,1,1),(1,1,2)} \\ \downarrow & & \downarrow & & \downarrow \\ \mathbb{R}[x,y]_2 & \times & \mathbb{R}[x^2,y^2]_1 & \longrightarrow & \mathcal{M}_{(2,1),(1,2)} \end{array}$$

Given an LCN architecture $(\boldsymbol{d}, \boldsymbol{S})$, merging neighboring layers with the same S_i yields an LCN architecture $(\tilde{\boldsymbol{d}}, \tilde{\boldsymbol{S}})$ with $1 = \tilde{S}_1 < \tilde{S}_2 < \tilde{S}_3 < \ldots$, called the reduced LCN architecture.

Singularities

Lemma: $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \subseteq \mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{S}}}$ and $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} = \overline{\mathcal{M}}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{S}}}$,
where $\bar{\cdot}$ denotes the Zariski closure inside $\mathbb{R}[x,y]_d$.

Singularities

Lemma: $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \subseteq \mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{S}}}$ and $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} = \overline{\mathcal{M}}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{S}}}$,
where $\bar{\cdot}$ denotes the Zariski closure inside $\mathbb{R}[x,y]_d$.

Theorem Let $(\boldsymbol{d}, \boldsymbol{S})$ be a reduced LCN architecture with L layers.

- If $L = 1$ (i.e., any associated non-reduced architecture has all strides equal 1), then $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} = \mathbb{R}[x,y]_d$.

Singularities

Lemma: $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \subseteq \mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{S}}}$ and $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} = \overline{\mathcal{M}}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{S}}}$,
where $\bar{\cdot}$ denotes the Zariski closure inside $\mathbb{R}[x,y]_d$.

Theorem Let $(\boldsymbol{d},\boldsymbol{S})$ be a reduced LCN architecture with L layers.

- If $L=1$ (i.e., any associated non-reduced architecture has all strides equal 1), then $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} = \mathbb{R}[x,y]_d$.
- If $L>1$, $\deg \overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} > 1$ and

$$\mathrm{Sing}(\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}}) = \{0\} \cup \bigcup_{d' \in D} \overline{\mathcal{M}}_{\boldsymbol{d'},\boldsymbol{S}} = \{0\} \cup \bigcup_{d' \in D} \mathcal{M}_{\boldsymbol{d'},\boldsymbol{S}},$$

where $D := \{\boldsymbol{d'} \in \mathbb{Z}^L_{\geq 0} \mid \overline{\mathcal{M}}_{\boldsymbol{d'},\boldsymbol{S}} \subsetneq \overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}}\}$

Singularities

Lemma: $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \subseteq \mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{S}}}$ and $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} = \overline{\mathcal{M}}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{S}}}$, where $\bar{\cdot}$ denotes the Zariski closure inside $\mathbb{R}[x,y]_d$.

Theorem Let $(\boldsymbol{d}, \boldsymbol{S})$ be a reduced LCN architecture with L layers.

- If $L = 1$ (i.e., any associated non-reduced architecture has all strides equal 1), then $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} = \mathbb{R}[x,y]_d$.
- If $L > 1$, $\deg \overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} > 1$ and

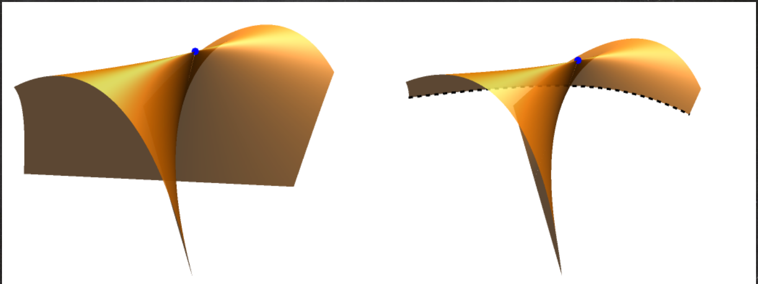
$$\mathrm{Sing}(\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}}) = \{0\} \cup \bigcup_{d' \in D} \overline{\mathcal{M}}_{\boldsymbol{d}',\boldsymbol{S}} = \{0\} \cup \bigcup_{d' \in D} \mathcal{M}_{\boldsymbol{d}',\boldsymbol{S}},$$

where $D := \{\boldsymbol{d}' \in \mathbb{Z}^L_{\geq 0} \mid \overline{\mathcal{M}}_{\boldsymbol{d}',\boldsymbol{S}} \subsetneq \overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}}\}$
$= \{\boldsymbol{d}' \in \mathbb{Z}^L_{\geq 0} \mid \boldsymbol{d}' \neq \boldsymbol{d}, \sum_{i=1}^{L} d'_i S_i = \sum_{i=1}^{L} d_i S_i, \forall l : \sum_{i=l}^{L} d'_i S_i \geq \sum_{i=l}^{L} d_i S_i\}$

Example

$\mu : \mathbb{R}[x, y]_2 \times \mathbb{R}[x^2, y^2]_1 \to \mathbb{R}[x, y]_4$

$\mu : \mathbb{R}[x, y]_1 \times \mathbb{R}[x, y]_1 \times \mathbb{R}[x^2, y^2]_1 \to \mathbb{R}[x, y]_4$

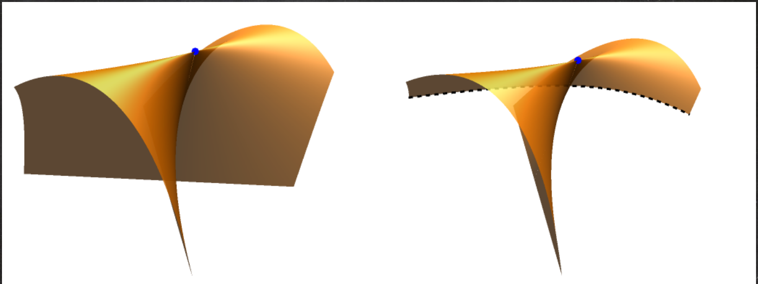


$\mathbb{R}[x, y]_2 \times \mathbb{R}[x^2, y^2]_1 \to \mathcal{M}_{(2,1),(1,2)}$

$\mathrm{Sing}(\overline{\mathcal{M}}_{(2,1),(1,2)}) =$

Example

$\mu : \mathbb{R}[x,y]_2 \times \mathbb{R}[x^2,y^2]_1 \to \mathbb{R}[x,y]_4$ $\qquad$ $\mu : \mathbb{R}[x,y]_1 \times \mathbb{R}[x,y]_1 \times \mathbb{R}[x^2,y^2]_1 \to \mathbb{R}[x,y]_4$



$\mathbb{R}[x,y]_2 \times \mathbb{R}[x^2,y^2]_1 \to \mathcal{M}_{(2,1),(1,2)}$

$\mathrm{Sing}(\overline{\mathcal{M}}_{(2,1),(1,2)}) = \mathcal{M}_{(0,2),(1,2)} = \mathbb{R}[x^2,y^2]_2$

Relative Boundary

$\partial\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} =$ points in $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$ that are limits of sequences in $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} \setminus \mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$.

Relative Boundary

$\partial\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$ = points in $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$ that are limits of sequences in $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} \setminus \mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$.

Recall: $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \subseteq \mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{S}}} \subseteq \overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{S}} = \overline{\mathcal{M}}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{S}}}$

Relative Boundary

$\partial\mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$ = points in $\mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$ that are limits of sequences in $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{s}} \setminus \mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$.

Recall: $\mathcal{M}_{\boldsymbol{d},\boldsymbol{s}} \subseteq \mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{s}}} \subseteq \overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{s}} = \overline{\mathcal{M}}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{s}}}$

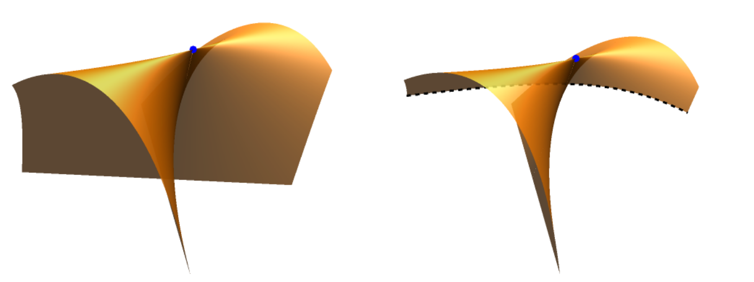
- reduced boundary points: limits in $\mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$ of sequences in $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{s}} \setminus \mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{s}}}$
- stride-1 boundary points: limits in $\mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$ of sequences in $\mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{s}}} \setminus \mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$

Relative Boundary

$\partial\mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$ = points in $\mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$ that are limits of sequences in $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{s}} \setminus \mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$.

Recall: $\mathcal{M}_{\boldsymbol{d},\boldsymbol{s}} \subseteq \mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{s}}} \subseteq \overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{s}} = \overline{\mathcal{M}}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{s}}}$

- reduced boundary points: limits in $\mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$ of sequences in $\overline{\mathcal{M}}_{\boldsymbol{d},\boldsymbol{s}} \setminus \mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{s}}}$
- stride-1 boundary points: limits in $\mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$ of sequences in $\mathcal{M}_{\tilde{\boldsymbol{d}},\tilde{\boldsymbol{s}}} \setminus \mathcal{M}_{\boldsymbol{d},\boldsymbol{s}}$



reduced boundary points have at least codimension 2
stride-1 boundary points (if existent) have codimension 1

Training with the squared error loss

Given training data $\mathcal{D} = \{(X_i, Y_i) \in \mathbb{R}^{k_0} \times \mathbb{R}^{k_L} \mid i = 1, \dots, N\}$, the squared error loss on the function space is

$$\begin{aligned} \ell_{\mathcal{D}} : \mathbb{R}^{k_L \times k_0} &\longrightarrow \mathbb{R}, \\ T &\longmapsto \sum_{i=1}^{N} \|Y_i - TX_i\|^2. \end{aligned}$$

Training with the squared error loss

Given training data $\mathcal{D} = \{(X_i, Y_i) \in \mathbb{R}^{k_0} \times \mathbb{R}^{k_L} \mid i = 1, \ldots, N\}$, the squared error loss on the function space is

$$\begin{aligned} \ell_{\mathcal{D}} : \mathbb{R}^{k_L \times k_0} &\longrightarrow \mathbb{R}, \\ T &\longmapsto \sum_{i=1}^{N} \|Y_i - TX_i\|^2. \end{aligned}$$

Training an LCN minimizes the squared error loss on the parameter space:

$$\begin{aligned} \mathcal{L}_{\mathcal{D}} : \mathbb{R}^{d_1} \times \ldots \times \mathbb{R}^{d_L} &\xrightarrow{\mu} \mathcal{M}_{\boldsymbol{d},\boldsymbol{s}} \subseteq \mathbb{R}^{k_L \times k_0} \xrightarrow{\ell_{\mathcal{D}}} \mathbb{R}, \\ (w_1, \ldots, w_L) &\longmapsto T_{w_L, s_L} \cdots T_{w_1, s_1} \longmapsto \ell_{\mathcal{D}}(T_{w_L, s_L} \cdots T_{w_1, s_1}) \end{aligned}$$

Training with the squared error loss

$$\mathcal{L}_{\mathcal{D}} : \mathbb{R}^{d_1} \times \ldots \times \mathbb{R}^{d_L} \xrightarrow{\mu} \mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \xrightarrow{\ell_{\mathcal{D}}} \mathbb{R}$$

Theorem

Let $(\boldsymbol{d}, \boldsymbol{S})$ be a reduced LCN architecture and let $N \geq \sum_i d_i S_i + 1$.
For almost all data $\mathcal{D} \in (\mathbb{R}^{k_0} \times \mathbb{R}^{k_L})^N$, every critical point $\boldsymbol{w}$ of $\mathcal{L}_{\mathcal{D}}$ satisfies one of the following:

Training with the squared error loss

$$\mathcal{L}_{\mathcal{D}} : \mathbb{R}^{d_1} \times \ldots \times \mathbb{R}^{d_L} \xrightarrow{\mu} \mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \xrightarrow{\ell_{\mathcal{D}}} \mathbb{R}$$

Theorem

Let $(\boldsymbol{d}, \boldsymbol{S})$ be a reduced LCN architecture and let $N \geq \sum_i d_i S_i + 1$.
For almost all data $\mathcal{D} \in (\mathbb{R}^{k_0} \times \mathbb{R}^{k_L})^N$, every critical point $\boldsymbol{w}$ of $\mathcal{L}_{\mathcal{D}}$ satisfies one of the following:

- $\mu(\boldsymbol{w}) = 0$, or
- $\boldsymbol{w}$ is a regular point of μ and $\mu(\boldsymbol{w})$ is a smooth, interior point of $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$

Training with the squared error loss

$$\mathcal{L}_{\mathcal{D}} : \mathbb{R}^{d_1} \times \ldots \times \mathbb{R}^{d_L} \xrightarrow{\mu} \mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \xrightarrow{\ell_{\mathcal{D}}} \mathbb{R}$$

Theorem

Let $(\boldsymbol{d}, \boldsymbol{S})$ be a reduced LCN architecture and let $N \geq \sum_i d_i S_i + 1$.
For almost all data $\mathcal{D} \in (\mathbb{R}^{k_0} \times \mathbb{R}^{k_L})^N$, every critical point $\boldsymbol{w}$ of $\mathcal{L}_{\mathcal{D}}$ satisfies one of the following:

- $\mu(\boldsymbol{w}) = 0$, or
- $\boldsymbol{w}$ is a regular point of μ and $\mu(\boldsymbol{w})$ is a smooth, interior point of $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$ (in particular, $\mu(\boldsymbol{w})$ is a critical point of $\ell_{\mathcal{D}}|_{\mathrm{Reg}(\mathcal{M}^{\circ}_{\boldsymbol{d},\boldsymbol{S}})}$).

Training with the squared error loss

$$\mathcal{L}_{\mathcal{D}} : \mathbb{R}^{d_1} \times \ldots \times \mathbb{R}^{d_L} \xrightarrow{\mu} \mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \xrightarrow{\ell_{\mathcal{D}}} \mathbb{R}$$

Theorem

Let $(\boldsymbol{d}, \boldsymbol{S})$ be a reduced LCN architecture and let $N \geq \sum_i d_i S_i + 1$.
For almost all data $\mathcal{D} \in (\mathbb{R}^{k_0} \times \mathbb{R}^{k_L})^N$, every critical point $\boldsymbol{w}$ of $\mathcal{L}_{\mathcal{D}}$ satisfies one of the following:

- $\mu(\boldsymbol{w}) = 0$, or
- $\boldsymbol{w}$ is a regular point of μ and $\mu(\boldsymbol{w})$ is a smooth, interior point of $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$ (in particular, $\mu(\boldsymbol{w})$ is a critical point of $\ell_{\mathcal{D}}|_{\mathrm{Reg}(\mathcal{M}^{\circ}_{\boldsymbol{d},\boldsymbol{S}})}$).

This is known to be **false** for

linear fully-connected networks	stride-1 LCNs

Training with the squared error loss

$$\mathcal{L}_\mathcal{D} : \mathbb{R}^{d_1} \times \ldots \times \mathbb{R}^{d_L} \xrightarrow{\mu} \mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \xrightarrow{\ell_\mathcal{D}} \mathbb{R}$$

Theorem

Let $(\boldsymbol{d}, \boldsymbol{S})$ be a reduced LCN architecture and let $N \geq \sum_i d_i S_i + 1$.
For almost all data $\mathcal{D} \in (\mathbb{R}^{k_0} \times \mathbb{R}^{k_L})^N$, every critical point $\boldsymbol{w}$ of $\mathcal{L}_\mathcal{D}$ satisfies one of the following:

- $\mu(\boldsymbol{w}) = 0$, or
- $\boldsymbol{w}$ is a regular point of μ and $\mu(\boldsymbol{w})$ is a smooth, interior point of $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$ (in particular, $\mu(\boldsymbol{w})$ is a critical point of $\ell_\mathcal{D}|_{\mathrm{Reg}(\mathcal{M}^\circ_{\boldsymbol{d},\boldsymbol{S}})}$).

This is known to be **false** for

linear fully-connected networks	**stride-1 LCNs**
$\mathcal{M}$ = determinantal variety	$\mathcal{M}$ = full-dimensional semi-algebraic

Training with the squared error loss

$$\mathcal{L}_{\mathcal{D}} : \mathbb{R}^{d_1} \times \ldots \times \mathbb{R}^{d_L} \xrightarrow{\mu} \mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \xrightarrow{\ell_{\mathcal{D}}} \mathbb{R}$$

Theorem
Let $(\boldsymbol{d}, \boldsymbol{S})$ be a reduced LCN architecture and let $N \geq \sum_i d_i S_i + 1$.
For almost all data $\mathcal{D} \in (\mathbb{R}^{k_0} \times \mathbb{R}^{k_L})^N$, every critical point $\boldsymbol{w}$ of $\mathcal{L}_{\mathcal{D}}$ satisfies one of the following:

- $\mu(\boldsymbol{w}) = 0$, or
- $\boldsymbol{w}$ is a regular point of μ and $\mu(\boldsymbol{w})$ is a smooth, interior point of $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$ (in particular, $\mu(\boldsymbol{w})$ is a critical point of $\ell_{\mathcal{D}}|_{\mathrm{Reg}(\mathcal{M}^\circ_{\boldsymbol{d},\boldsymbol{S}})}$).

This is known to be **false** for

linear fully-connected networks	**stride-1 LCNs**
$\mathcal{M} =$ determinantal variety	$\mathcal{M} =$ full-dimensional semi-algebraic
critical points are often on $\mathrm{Sing}(\mathcal{M})$	critical points are often on $\partial\mathcal{M}$

Training with the squared error loss

$$\mathcal{L}_\mathcal{D} : \mathbb{R}^{d_1} \times \ldots \times \mathbb{R}^{d_L} \xrightarrow{\mu} \mathcal{M}_{\boldsymbol{d},\boldsymbol{S}} \xrightarrow{\ell_\mathcal{D}} \mathbb{R}$$

Theorem

Let $(\boldsymbol{d}, \boldsymbol{S})$ be a reduced LCN architecture and let $N \geq \sum_i d_i S_i + 1$.
For almost all data $\mathcal{D} \in (\mathbb{R}^{k_0} \times \mathbb{R}^{k_L})^N$, every critical point $\boldsymbol{w}$ of $\mathcal{L}_\mathcal{D}$ satisfies one of the following:

- $\mu(\boldsymbol{w}) = 0$, or
- $\boldsymbol{w}$ is a regular point of μ and $\mu(\boldsymbol{w})$ is a smooth, interior point of $\mathcal{M}_{\boldsymbol{d},\boldsymbol{S}}$ (in particular, $\mu(\boldsymbol{w})$ is a critical point of $\ell_\mathcal{D}|_{\mathrm{Reg}(\mathcal{M}^\circ_{\boldsymbol{d},\boldsymbol{S}})}$).

<u>This is known to be **false** for</u>

linear fully-connected networks	**stride-1 LCNs**
$\mathcal{M}$ = determinantal variety	$\mathcal{M}$ = full-dimensional semi-algebraic
critical points are often on $\mathrm{Sing}(\mathcal{M})$	critical points are often on $\partial\mathcal{M}$

critical points are often "spurious", i.e. $\mu(\boldsymbol{w}) \notin \mathrm{Crit}(\ell_\mathcal{D}|\mathcal{M})$