NEURAL INFORMATION PROCESSING SYSTEMS

Abide by the law and follow the flow: Conservation laws for gradient flows

Sibylle Marcotte

ENS
ÉCOLE NORMALE SUPÉRIEURE
cnrs
ENS DE LYON
Inria

Context

Empirical risk minimization:

$$\mathcal{E}_{X,Y}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \ell\left(g(\theta, x_i), y_i\right)$$

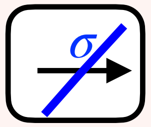
Context

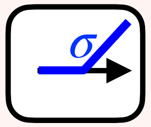
Empirical risk minimization:

$$\mathscr{E}_{X,Y}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \ell\left(g(\theta, x_i), y_i\right)$$

2-layer Neural Network: $\theta = (U, V)$

$$g(\theta, x) = U\sigma(V^\top x) = \sum_k u_k \sigma(\langle x, v_k \rangle)$$



Linear case or


ReLU case

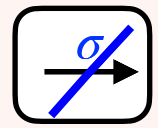
Context

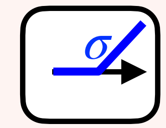
Empirical risk minimization:

$$\mathcal{E}_{X,Y}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \ell\left(g(\theta, x_i), y_i\right)$$

2-layer Neural Network: $\theta = (U, V)$

$$g(\theta, x) = U\sigma(V^\top x) = \sum_k u_k \sigma(\langle x, v_k \rangle)$$



Linear case or


ReLU case

Gradient flow:

$$\dot{\theta}(t) = -\nabla \mathcal{E}_{X,Y}(\theta(t)), \;\; \theta(0) = \theta_0$$

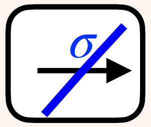
Context

Empirical risk minimization:

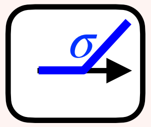
$$\mathscr{E}_{X,Y}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \ell\left(g(\theta, x_i), y_i\right)$$

2-layer Neural Network: $\theta = (U, V)$

$$g(\theta, x) = U\sigma(V^\top x) = \sum_k u_k \sigma(\langle x, v_k \rangle)$$



Linear case or



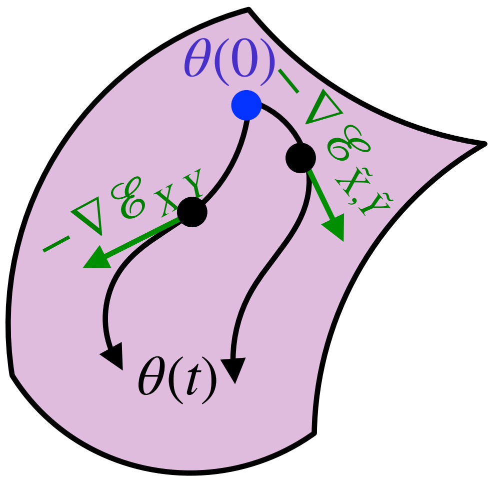
ReLU case

Gradient flow:

$$\dot{\theta}(t) = -\nabla \mathscr{E}_{X,Y}(\theta(t)), \ \ \theta(0) = \theta_0$$

Definition: Conserved functions

$$h(\theta(t)) = h(\theta_0), \forall \theta_0, X, Y, t$$



$\{\theta : h(\theta) = h(\theta(0))\}$

Main question: what are the conserved functions, how many are they?

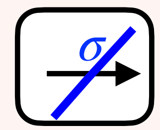
Context

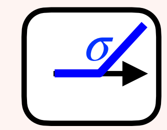
Empirical risk minimization:

$$\mathcal{E}_{X,Y}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \ell\left(g(\theta, x_i), y_i\right)$$

2-layer Neural Network: $\theta = (U, V)$

$$g(\theta, x) = U\sigma(V^\top x) = \sum_k u_k \sigma(\langle x, v_k \rangle)$$



Linear case or


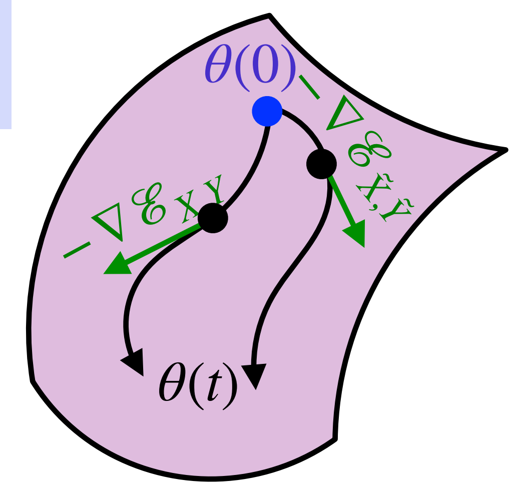
ReLU case

Gradient flow:

$$\dot{\theta}(t) = -\nabla \mathcal{E}_{X,Y}(\theta(t)), \;\; \theta(0) = \theta_0$$

Definition: Conserved functions

$$h(\theta(t)) = h(\theta_0), \forall \theta_0, X, Y, t$$



$\{\theta : h(\theta) = h(\theta(0))\}$

Main question: what are the conserved functions, how many are they?

Applications:

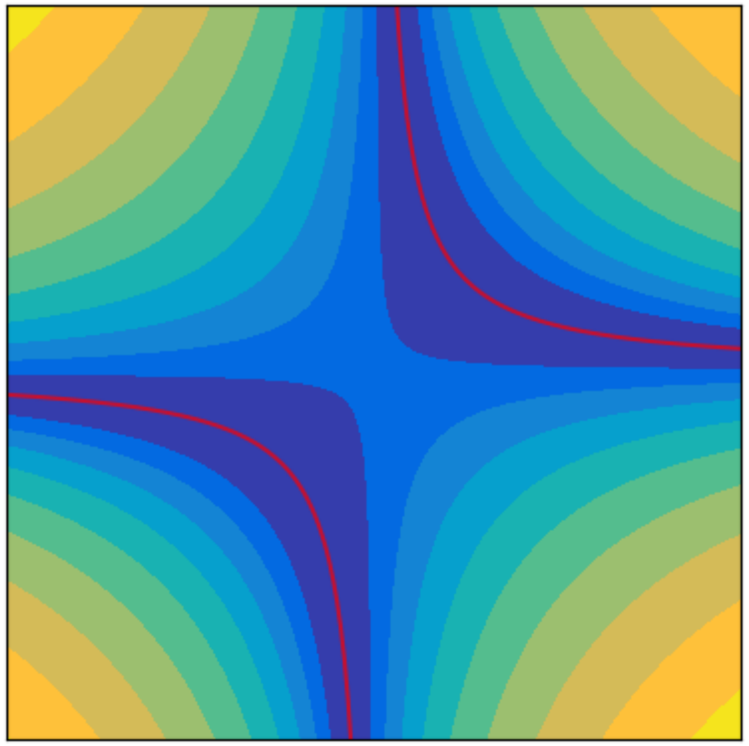
- Understanding implicit bias of gradient descent.
- Helping to analyze convergence.

Objectives

Example: 1D linear network

$\theta = (u, v)$, $g(\theta, x) = uvx$.

$\mathscr{E}_{X,Y}(u, v) = (uvx - y)^2$

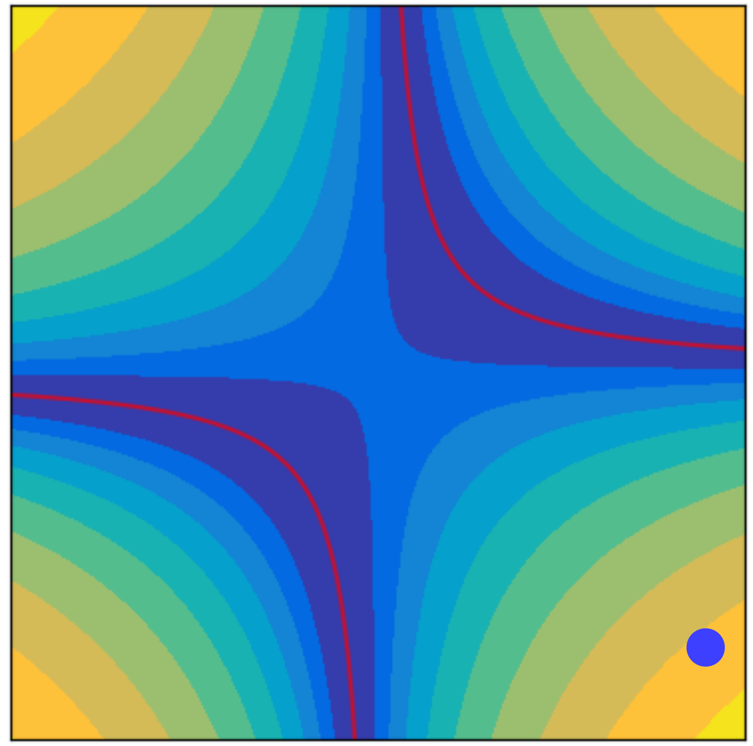


Objectives

Example: 1D linear network

$\theta = (u, v)$, $g(\theta, x) = uvx$.

$\mathscr{E}_{X,Y}(u, v) = (uvx - y)^2$

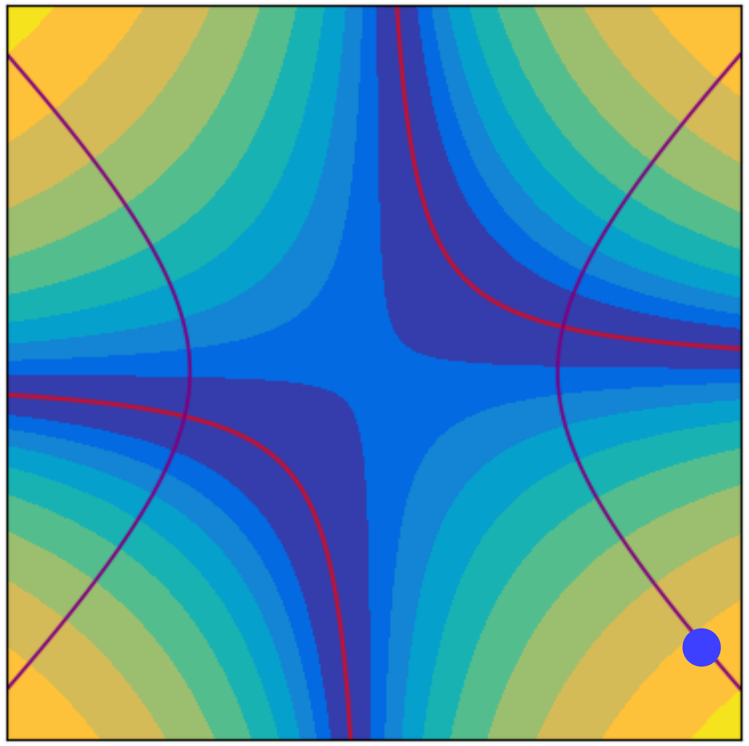


Objectives

Example: 1D linear network

$\theta = (u, v)$, $g(\theta, x) = uvx$.

$\mathcal{E}_{X,Y}(u, v) = (uvx - y)^2$



A conserved function:

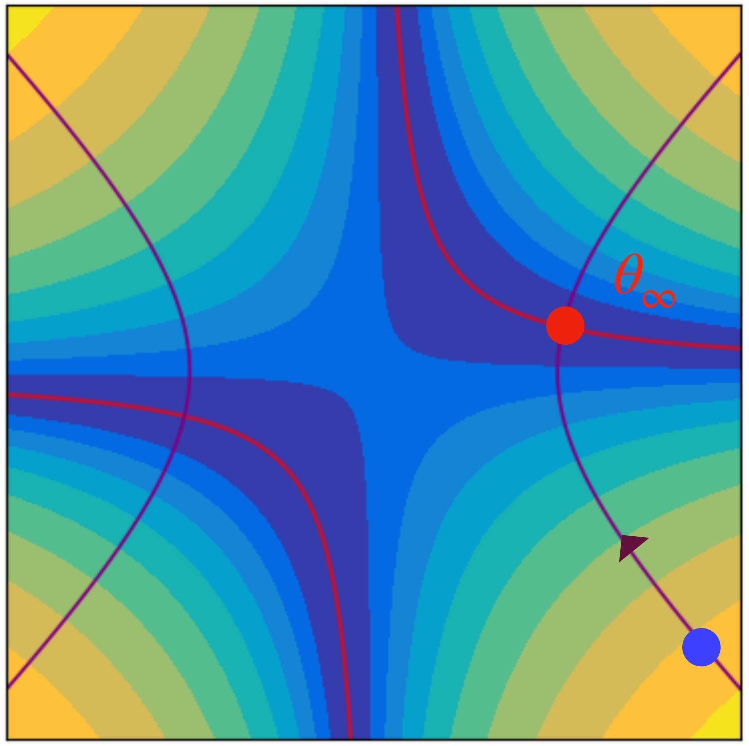
$h(\theta) = u^2 - v^2 = u_0^2 - v_0^2$

Objectives

Example: 1D linear network

$\theta = (u, v)$, $g(\theta, x) = uvx$.

$\mathscr{E}_{X,Y}(u, v) = (uvx - y)^2$



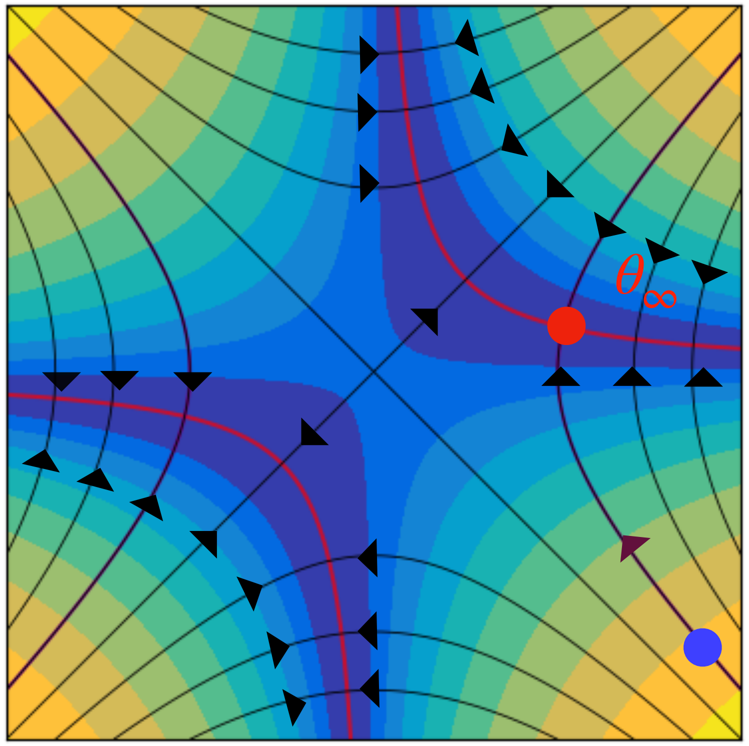
A conserved function:

$h(\theta) = u^2 - v^2 = u_0^2 - v_0^2$

Example: 1D linear network

$\theta = (u, v)$, $g(\theta, x) = uvx$.

$\mathscr{E}_{X,Y}(u, v) = (uvx - y)^2$



A conserved function:

$h(\theta) = u^2 - v^2 = u_0^2 - v_0^2$

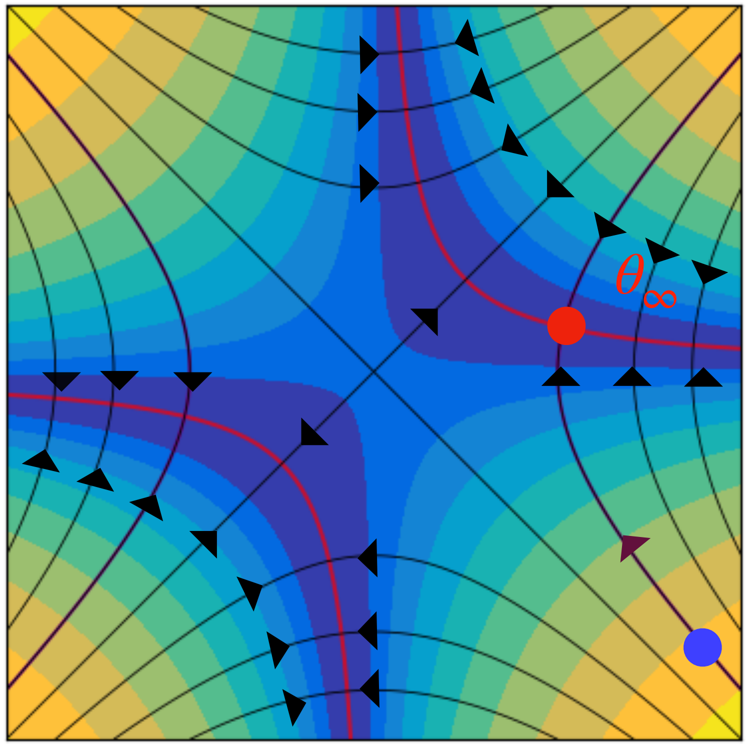
Objectives

Example: 1D linear network

$\theta = (u, v)$, $g(\theta, x) = uvx$.

$\mathscr{E}_{X,Y}(u, v) = (uvx - y)^2$

$uvx = y$



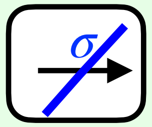
$\{\theta : h(\theta) = h(\theta_0)\}$

A conserved function:

$h(\theta) = u^2 - v^2 = u_0^2 - v_0^2$

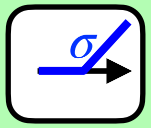
$\theta_0 = (u_0, v_0)$

Linear networks [1]



$h_{k,l}(U, V) = \langle u_k, u_l \rangle - \langle v_k, v_l \rangle$

ReLu networks [2]



$h_k(U, V) = \|u_k\|^2 - \|v_k\|^2$

[1] Arora et al. On the optimization of deep networks: Implicit acceleration by over-parameterization, ICML, 2018

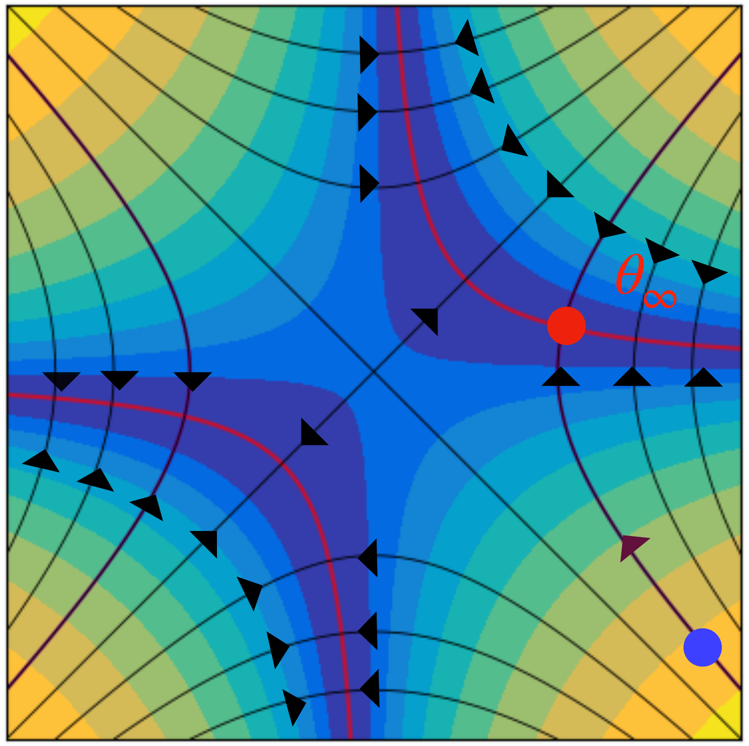
[2] Du et al. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced, Neurips, 2018

Objectives

Example: 1D linear network

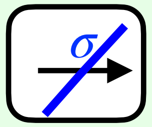
$\theta = (u, v)$, $g(\theta, x) = uvx$.

$\mathscr{E}_{X,Y}(u, v) = (uvx - y)^2$

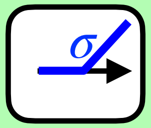


A conserved function:

$h(\theta) = u^2 - v^2 = u_0^2 - v_0^2$

Linear networks [1] 

$h_{k,l}(U, V) = \langle u_k, u_l \rangle - \langle v_k, v_l \rangle$

ReLu networks [2] 

$h_k(U, V) = \|u_k\|^2 - \|v_k\|^2$

Goal: Find a maximal set of 'independent' conserved functions

$(h_1, \ldots, h_K)$ conserved $\Rightarrow \Phi(h_1, \ldots, h_K)$ conserved

Definition of independence: $(\nabla h_i(\theta))_i$ linearly independent $\forall \theta$

[1] Arora et al. On the optimization of deep networks: Implicit acceleration by over-parameterization, ICML, 2018
[2] Du et al. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced, Neurips, 2018

Problem abstraction

$\dot{\theta}(t) = w(\theta(t))$ with vector field $w(\,\cdot\,)$, $w(\theta) \in W(\theta)$

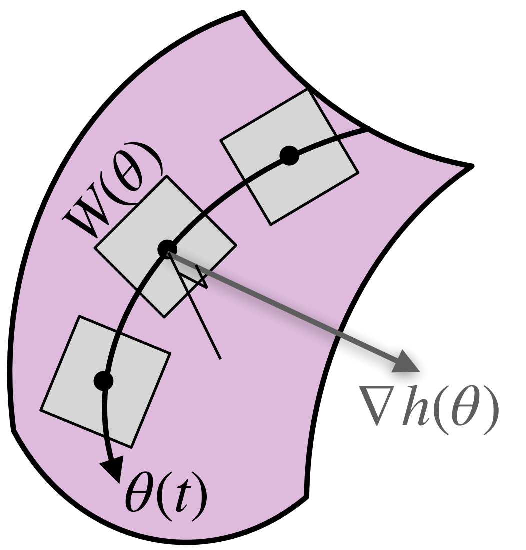
Definition: $W(\theta) := \text{Span}\left\{ \nabla \mathscr{E}_{X,Y}(\theta) \ : \ X, Y \right\}$

Problem abstraction

$\dot{\theta}(t) = w(\theta(t))$ with vector field $w(\cdot)$, $w(\theta) \in W(\theta)$

Definition: $W(\theta) := \text{Span}\left\{ \nabla \mathcal{E}_{X,Y}(\theta) \ : \ X, Y \right\}$

Proposition: h conserved $\Leftrightarrow \forall \theta, \nabla h(\theta) \perp W(\theta)$

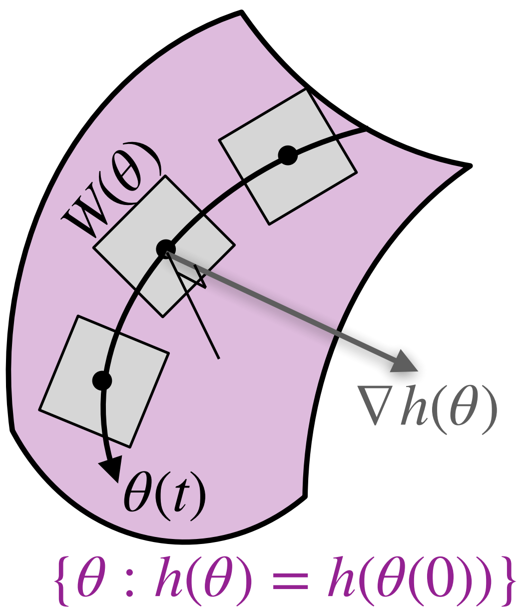


Problem abstraction

$\dot{\theta}(t) = w(\theta(t))$ with vector field $w(\cdot)$, $w(\theta) \in W(\theta)$

Definition: $W(\theta) := \text{Span}\left\{\nabla \mathscr{E}_{X,Y}(\theta) \ : \ X, Y\right\}$

Proposition: h conserved $\Leftrightarrow \forall \theta, \nabla h(\theta) \perp W(\theta)$



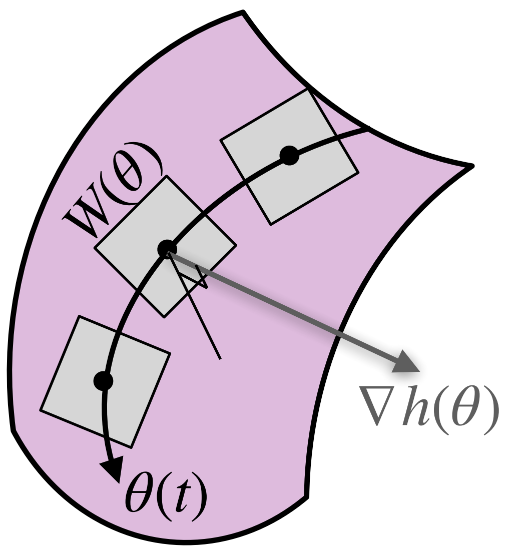
Question: Simpler expression of $W(\theta)$ to *find* conserved functions?

Problem abstraction

$\dot{\theta}(t) = w(\theta(t))$ with vector field $w(\cdot)$, $w(\theta) \in W(\theta)$

Definition: $W(\theta) := \text{Span}\left\{ \nabla \mathscr{E}_{X,Y}(\theta) \ : \ X, Y \right\}$

Proposition: h conserved $\Leftrightarrow \forall \theta, \nabla h(\theta) \perp W(\theta)$

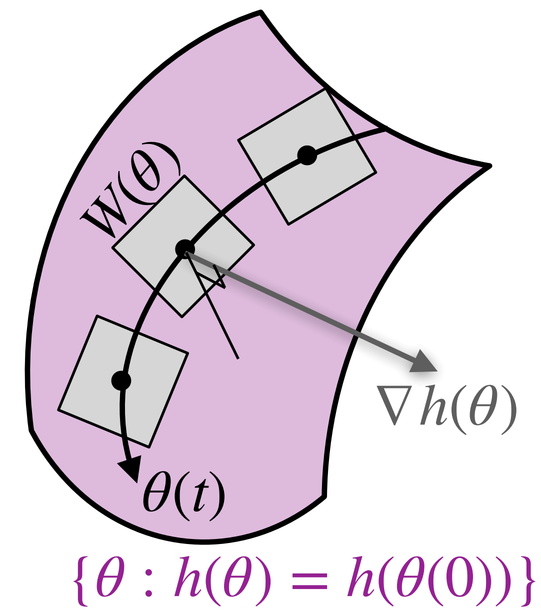


Question: Simpler expression of $W(\theta)$ to *find* conserved functions?

Tool: Re-parametrization: $g(\theta, x) = f(\varphi(\theta), x)$

Problem abstraction

$\dot{\theta}(t) = w(\theta(t))$ with vector field $w(\cdot)$, $w(\theta) \in W(\theta)$



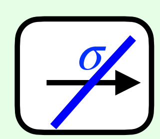
Definition: $W(\theta) := \text{Span}\left\{ \nabla \mathscr{E}_{X,Y}(\theta) \; : \; X, Y \right\}$

Proposition: h conserved $\Leftrightarrow \forall \theta, \nabla h(\theta) \perp W(\theta)$

Question: Simpler expression of $W(\theta)$ to *find* conserved functions?

Tool: Re-parametrization: $g(\theta, x) = f(\varphi(\theta), x)$

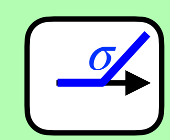
Linear networks



$$g(\theta, x) = UV^\top x$$

$$\varphi(U, V) = UV^\top$$

ReLu networks



$$g(\theta, x) = \sum_i u_i \text{ReLu}\left(\langle v_i, x\rangle\right) = \sum_i \mathbf{1}_{\langle v_i, x\rangle \geq 0}\left(u_i v_i^\top\right) x$$

$$\varphi(U, V) = (u_i v_i^\top)_i$$

From conserved function to conservation law

$$\mathscr{E}_{X,Y}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \ell\left(g(\theta, x_i), y_i\right) \qquad W(\theta) := \mathrm{Span}\left\{\nabla \mathscr{E}_{X,Y}(\theta) \ : \ X, Y\right\} \qquad g(\theta, x) = f(\varphi(\theta), x)$$

Chain rules

Proposition: $W(\theta) = \partial\varphi(\theta)^\top \, \mathrm{Span}\{\partial f(\varphi(\theta), x)^\top \nabla \ell(g(\theta, x), y) \ : \ X, Y\}$

finite-dimensional $\subseteq W_\varphi(\theta) := \mathrm{range}\left(\partial\varphi(\theta)^\top\right)$

From conserved function to conservation law

$$\mathcal{E}_{X,Y}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \ell\left(g(\theta, x_i), y_i\right) \qquad W(\theta) := \text{Span}\left\{\nabla \mathcal{E}_{X,Y}(\theta) \;:\; X, Y\right\} \qquad g(\theta, x) = f(\varphi(\theta), x)$$

Chain rules

Proposition: $W(\theta) = \partial\varphi(\theta)^\top \, \text{Span}\{\partial f(\varphi(\theta), x)^\top \nabla \ell(g(\theta, x), y) \;:\; X, Y\}$

finite-dimensional $\subseteq W_\varphi(\theta) := \text{range}\left(\partial\varphi(\theta)^\top\right)$

Theorem: $W(\theta) = W_\varphi(\theta)$ if $\text{Span}_y \nabla \ell(z, y) =$ whole space, for linear and 2-layer ReLU networks.

From conserved function to conservation law

$$\mathcal{E}_{X,Y}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \ell\left(g(\theta, x_i), y_i\right) \qquad W(\theta) := \text{Span}\left\{\nabla \mathcal{E}_{X,Y}(\theta) \;:\; X, Y\right\} \qquad g(\theta, x) = f(\varphi(\theta), x)$$

Chain rules

Proposition: $W(\theta) = \partial\varphi(\theta)^\top \, \text{Span}\{\partial f(\varphi(\theta), x)^\top \nabla \ell(g(\theta, x), y) \;:\; X, Y\}$

finite-dimensional $\subseteq W_\varphi(\theta) := \text{range}\left(\partial\varphi(\theta)^\top\right)$

Theorem: $W(\theta) = W_\varphi(\theta)$ if $\text{Span}_y \nabla \ell(z, y) =$ whole space, for linear and 2-layer ReLU networks.

Consequence: under the same hypothesis,

h conserved $\Leftrightarrow \forall \theta, \nabla h(\theta) \perp W_\varphi(\theta)$

Definition: Conservation law h

$\forall \theta, \nabla h(\theta) \perp W_\varphi(\theta)$

From conserved function to conservation law

$$\mathcal{E}_{X,Y}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \ell\left(g(\theta, x_i), y_i\right) \qquad W(\theta) := \text{Span}\left\{\nabla \mathcal{E}_{X,Y}(\theta) \;:\; X, Y\right\} \qquad g(\theta, x) = f(\varphi(\theta), x)$$

Chain rules

Proposition: $W(\theta) = \partial\varphi(\theta)^\top \, \text{Span}\{\partial f(\varphi(\theta), x)^\top \nabla \ell(g(\theta, x), y) \;:\; X, Y\}$

finite-dimensional $\subseteq W_\varphi(\theta) := \text{range}\left(\partial\varphi(\theta)^\top\right)$

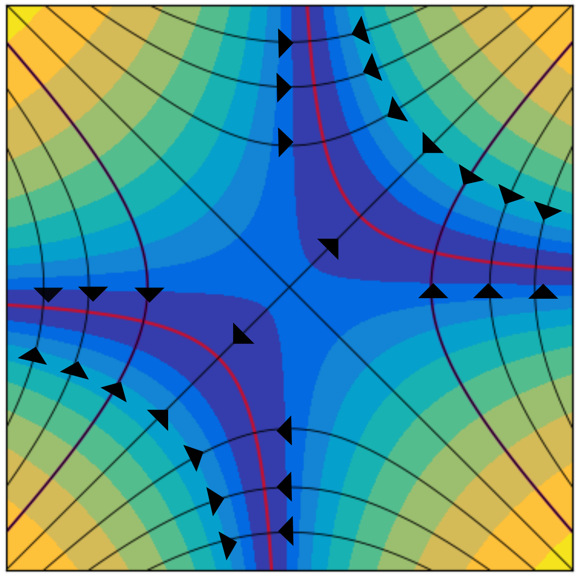
Theorem: $W(\theta) = W_\varphi(\theta)$ if $\text{Span}_y \nabla \ell(z, y) =$ whole space, for linear and 2-layer ReLU networks.

Consequence: under the same hypothesis,
h conserved $\Leftrightarrow \forall \theta, \nabla h(\theta) \perp W_\varphi(\theta)$

Definition: Conservation law h
$\forall \theta, \nabla h(\theta) \perp W_\varphi(\theta)$

Example: 1D example
$\theta = (u, v)$
$g(\theta, x) = uvx$
$\varphi(u, v) = uv$

$uvx = y$



$$\nabla h(\theta) = \begin{pmatrix} 2u \\ -2v \end{pmatrix} \perp \begin{pmatrix} v \\ u \end{pmatrix} = \partial\varphi(\theta)^\top$$

$$h(\theta) = u^2 - v^2 = u_0^2 - v_0^2$$

We can build some conservation laws

Consequence: as h conservation law $\Leftrightarrow \partial\varphi(\theta)\nabla h(\theta) \equiv 0$ (linear in h)

For a polynomial φ, restricting to polynomials h: finite dimensional linear kernel

We can build some conservation laws

Consequence: as h conservation law $\Leftrightarrow \partial\varphi(\theta)\nabla h(\theta) \equiv 0$ (linear in h)

For a polynomial φ, restricting to polynomials h: finite dimensional linear kernel

👩🏻‍💻 **ALGO<1>** returns all *polynomial* independent conservation laws

→ lower bound on number of all (*polynomial or not)* independent conservation laws

We can build some conservation laws

Consequence: as h conservation law $\Leftrightarrow \partial\varphi(\theta)\,\nabla h(\theta) \equiv 0$ (linear in h)

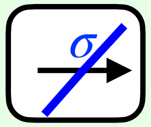
For a polynomial φ, restricting to polynomials h: finite dimensional linear kernel

ALGO<1> returns all *polynomial* independent conservation laws

→ lower bound on number of all (*polynomial or not)* independent conservation laws

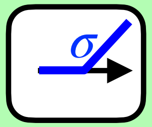
→ finds back all already known conserved functions [1, 2]:

Linear networks



$h_{k,l}(U, V) = \langle u_k, u_l \rangle - \langle v_k, v_l \rangle$

ReLu networks



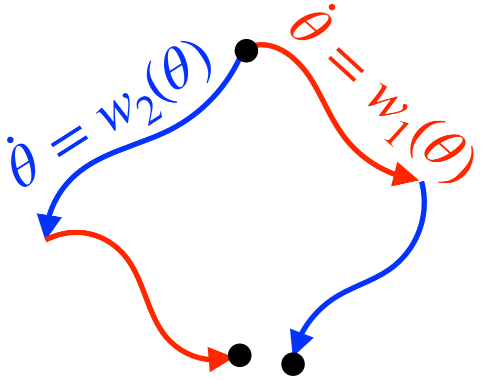
$h_k(U, V) = \|u_k\|^2 - \|v_k\|^2$

[1] Arora et al. On the optimization of deep networks: Implicit acceleration by over-parameterization, ICML, 2018
[2] Du et al. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced, Neurips, 2018

Conservation laws using Lie algebra

Definition: Lie brackets

$[w_1, w_2](\theta) := \partial w_1(\theta) w_2(\theta) - \partial w_2(\theta) w_1(\theta)$



$=$ if $[w_1, w_2] = 0$

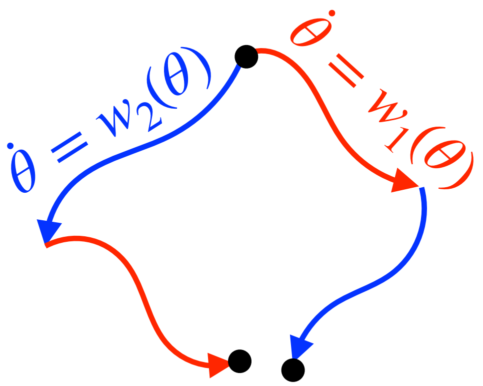
Conservation laws using Lie algebra

Definition: Lie brackets

$[w_1, w_2](\theta) := \partial w_1(\theta) w_2(\theta) - \partial w_2(\theta) w_1(\theta)$

Definition: Generated Lie algebra Lie(W_φ)

smallest space $\supset W_\varphi$ stable by $[\,\cdot\,,\cdot\,]$

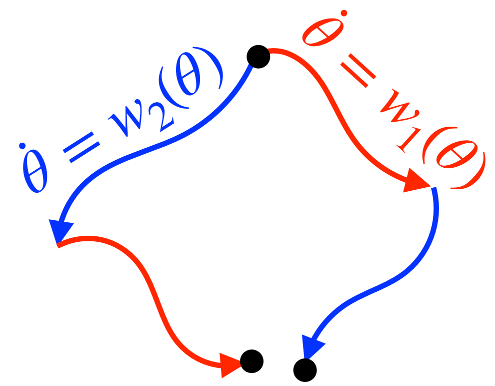


$=$ if $[w_1, w_2] = 0$

Conservation laws using Lie algebra

Definition: Lie brackets
$[w_1, w_2](\theta) := \partial w_1(\theta) w_2(\theta) - \partial w_2(\theta) w_1(\theta)$

Definition: Generated Lie algebra $\text{Lie}(W_\varphi)$
smallest space $\supset W_\varphi$ stable by $[\,\cdot\,,\cdot\,]$



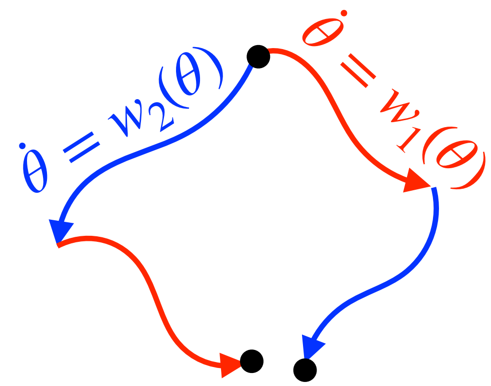
$=$ if $[w_1, w_2] = 0$

Theorem: If $\dim \left(\text{Lie}(W_\varphi)(\theta)\right) = K$ is locally constant,
there are *exactly* $D - K$ independent conservation laws

Conservation laws using Lie algebra

Definition: Lie brackets

$[w_1, w_2](\theta) := \partial w_1(\theta) w_2(\theta) - \partial w_2(\theta) w_1(\theta)$



$=$ if $[w_1, w_2] = 0$

Definition: Generated Lie algebra Lie(W_φ)

smallest space $\supset W_\varphi$ stable by $[\cdot, \cdot]$

Computationally tractable

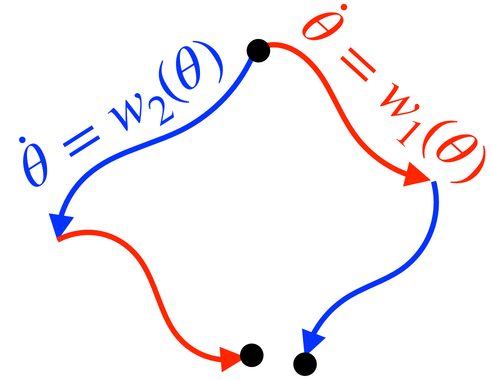
Theorem: If $\dim \Big(\text{Lie}(W_\varphi)(\theta)\Big) = K$ is locally constant,

there are *exactly* $D - K$ independent conservation laws

ALGO<2> returns dim Lie$(W_\phi)(\theta)$

Definition: Lie brackets
$[w_1, w_2](\theta) := \partial w_1(\theta) w_2(\theta) - \partial w_2(\theta) w_1(\theta)$

Definition: Generated Lie algebra Lie(W_φ)
smallest space $\supset W_\varphi$ stable by $[\cdot, \cdot]$



$=$ if $[w_1, w_2] = 0$

Computationally tractable

Theorem: If dim $\left(\text{Lie}(W_\varphi)(\theta)\right)$ = K is locally constant,
there are *exactly* $D - K$ independent conservation laws

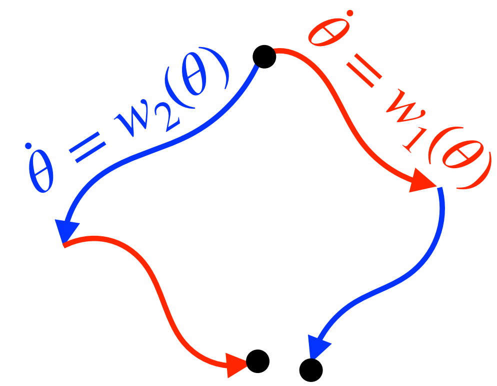
ALGO<2> returns dim Lie$(W_\phi)(\theta)$

Question: Have we found all conservation laws?

Conservation laws using Lie algebra

Definition: Lie brackets
$[w_1, w_2](\theta) := \partial w_1(\theta) w_2(\theta) - \partial w_2(\theta) w_1(\theta)$

Definition: Generated Lie algebra $\mathrm{Lie}(W_\varphi)$
smallest space $\supset W_\varphi$ stable by $[\cdot,\cdot]$



$=$ if $[w_1, w_2] = 0$

Computationally tractable

Theorem: If $\dim\left(\mathrm{Lie}(W_\varphi)(\theta)\right) = K$ is locally constant,
there are *exactly* $D - K$ independent conservation laws

ALGO<2> returns $\dim \mathrm{Lie}(W_\phi)(\theta)$

Question: Have we found all conservation laws?

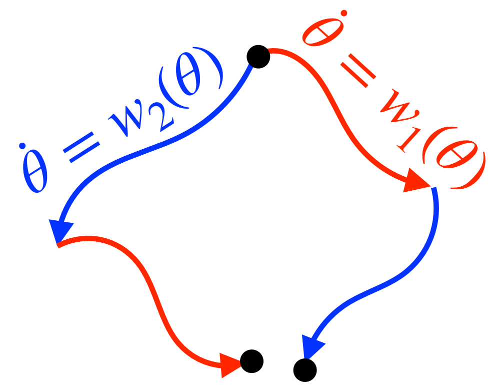
Proposition (for linear and ReLU networks):

- 2-layer case: analytic characterization of $\mathrm{Lie} W_\varphi$, $\mathrm{Lie} W_\varphi(\theta)$ and $\dim \mathrm{Lie} W_\varphi(\theta)$
- Deeper cases: numerical comparison with **ALGO<1>** and **ALGO<2>**

Conservation laws using Lie algebra

Definition: Lie brackets
$[w_1, w_2](\theta) := \partial w_1(\theta) w_2(\theta) - \partial w_2(\theta) w_1(\theta)$

Definition: Generated Lie algebra Lie(W_φ)
smallest space $\supset W_\varphi$ stable by $[\,\cdot\,,\cdot\,]$



$=$ if $[w_1, w_2] = 0$

Computationally tractable

Theorem: If dim $\left(\text{Lie}(W_\varphi)(\theta)\right)$ = K is locally constant,
there are *exactly* $D - K$ independent conservation laws

ALGO<2> returns dim Lie$(W_\phi)(\theta)$

Question: Have we found all conservation laws?

Proposition (for linear and ReLU networks):

- 2-layer case: analytic characterization of LieW_φ, Lie$W_\varphi(\theta)$ and dimLie$W_\varphi(\theta)$
- Deeper cases: numerical comparison with **ALGO<1>** and **ALGO<2>**

→ no other conservation laws

Conclusion

- Algorithm that builds polynomial conservation laws: lower bound
- Number of independent laws characterized by a Lie algebra
- Algorithm that computes this number

sage `https://github.com/sibyllema/Conservation_laws`

Poster: Great Hall & Hall B1+B2 (level 1) #900

Follow the flow … and

Follow @SibylleMarcotte :)

Paper pdf