Linear Separability and Geometry of Neural Representations

Cengiz ("Jen-ghiz") Pehlevan

Applied Mathematics & Kempner Institute

Harvard University

Harvard John A. Paulson School of Engineering and Applied Sciences

Kempner Institute For the Study of Natural & Artificial Intelligence at Harvard University

- We can impose principles on data representations

- We can extract principles from learned data representations

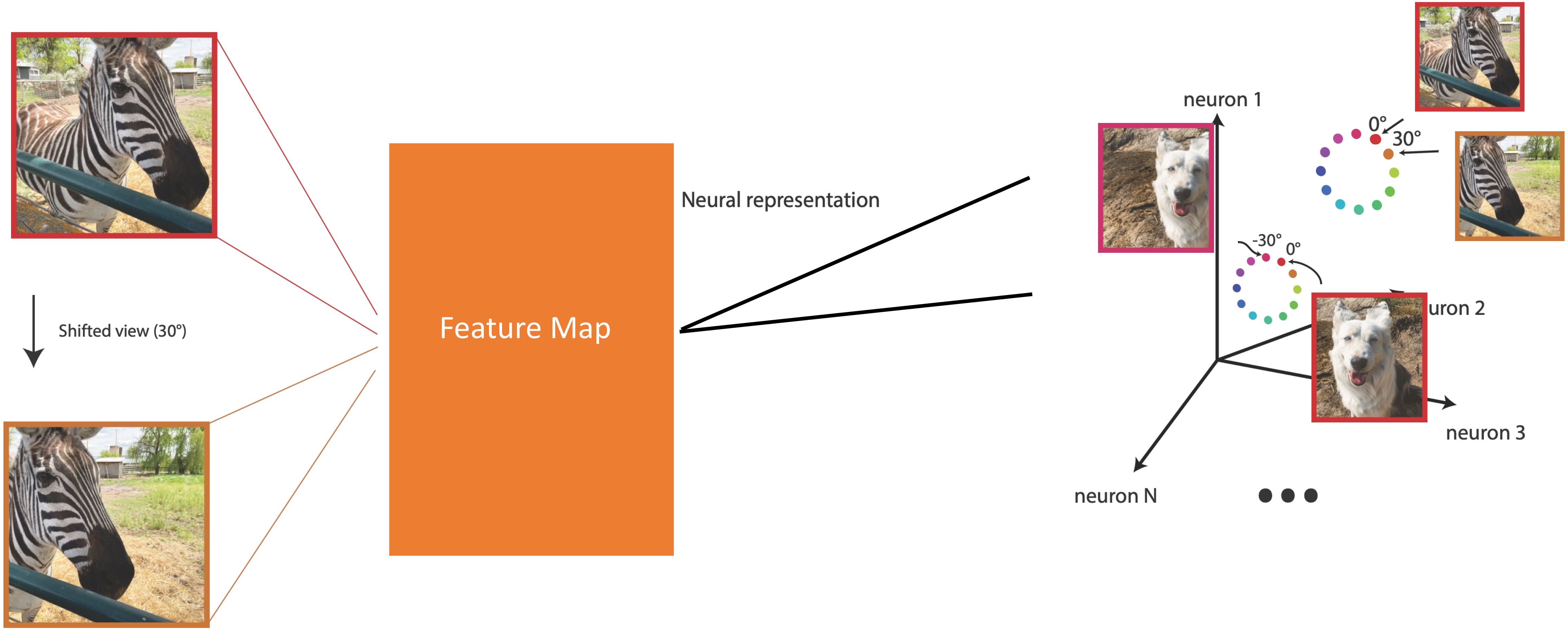
Capacity of Group-invariant Linear Readouts from Equivariant Representations: How Many Objects can be Linearly Classified Under All Possible Views?

Matthew Farrell[*‡], **Blake Bordelon**[*‡], **Shubhendu Trivedi**[†], **& Cengiz Pehlevan**[‡]

[‡] Harvard University
{msfarrell,blake_bordelon,cpehlevan}@seas.harvard.edu

[†] Massachusetts Institute of Technology
shubhendu@csail.mit.edu

- Equivariant representations are awesome!
- Equivariance is a strong constraint on representations.
- How does equivariance affect the expressivity of models?



Question: Assign random (+1, -1) labels to each “object orbit”. What fraction of such assignments are realizable by a linear classifier?

Notation and Setup

$\mathbf{x}$	Object
$g \in G$	Group acting on $\mathbf{x}$
$\mathbf{r}(\mathbf{x}) \in \mathbb{R}^N$	Feature map (can be anything)
$\pi : G \to GL(\mathbb{R}^N)$	Matrix representation $\quad \mathbf{r}(g\mathbf{x}) = \pi(g)\mathbf{r}(\mathbf{x})$
$\mathbf{x}^\mu, \quad \mu = 1, \ldots, P$	*P* objects
$\{\pi(g)\mathbf{r}(\mathbf{x}^\mu) : g \in G\}$	*P* orbits or manifolds
$y^\mu \in \{-1, 1\}$	labels

Dichotomy is linearly separable is there exists a $\mathbf{w}$ such that $y^\mu \mathbf{w}^\top \pi(g)\mathbf{r}(\mathbf{x}^\mu) > 0$ for all $g \in G$ and $\mu \in [P]$

Perceptron Capacity and Cover's counting theorem (c.f. VC Dimension)

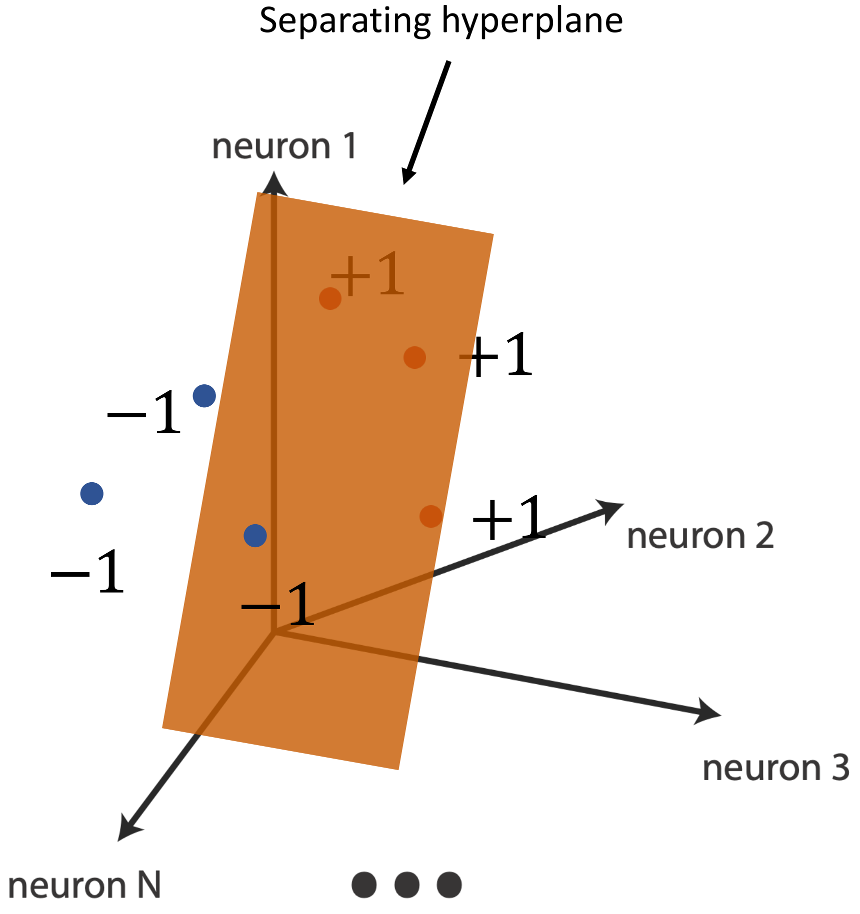
- Perceptron capacity — the fraction of possible labelings (+1 or -1) of points such that there exists a hyperplane passing through the origin where every +1 point is on one side and every -1 point is on the other.

- For P points (in general position) this is known and is given by

$$f(P,N) = 2^{1-P} \sum_{k=0}^{N-1} \binom{P-1}{k}$$

where N is the number of neurons/dimensions.

Cover, Thomas M. "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition." IEEE Transactions on Electronic Computers (1965)



Perceptron Capacity and Cover's counting theorem (c.f. VC Dimension)

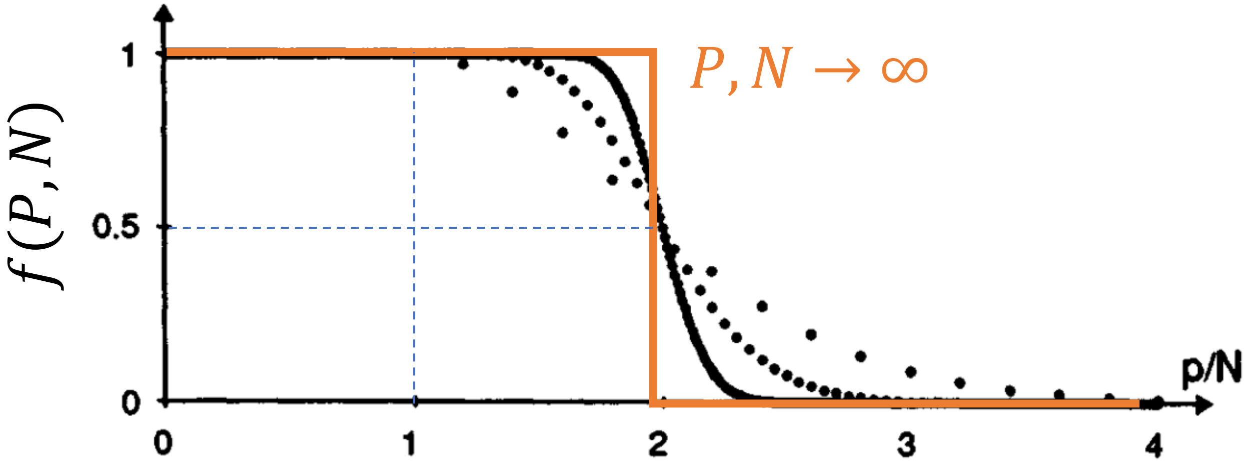
- Perceptron capacity — the fraction of possible labelings (+1 or -1) of points such that there exists a hyperplane passing through the origin where every +1 point is on one side and every -1 point is on the other.

- For P points (in general position) this is known and is given by

$$f(P,N) = 2^{1-P} \sum_{k=0}^{N-1} \binom{P-1}{k}$$

where N is the number of neurons/dimensions.

- $P = N$ is the VC (shattering) dimension

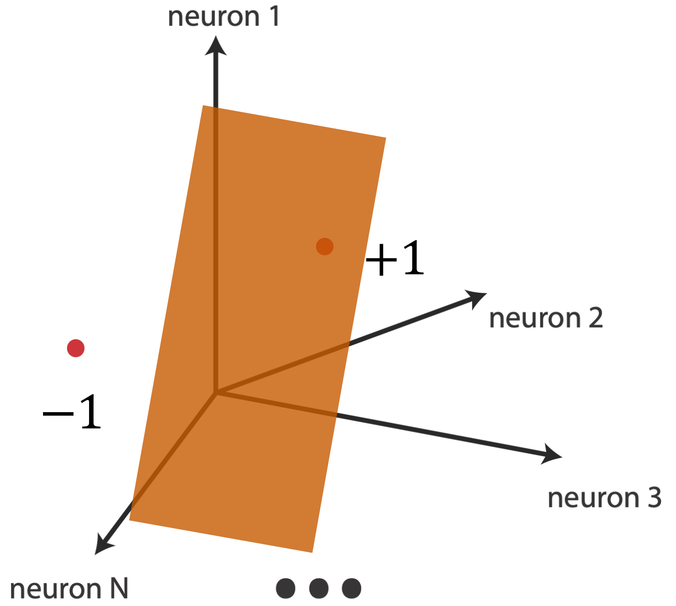


$N = 5, 20, 100$

Main Result

<u>Cover's theorem:</u>

$$f(P,N) = 2^{1-P}\sum_{k=0}^{N-1}\binom{P-1}{k}$$

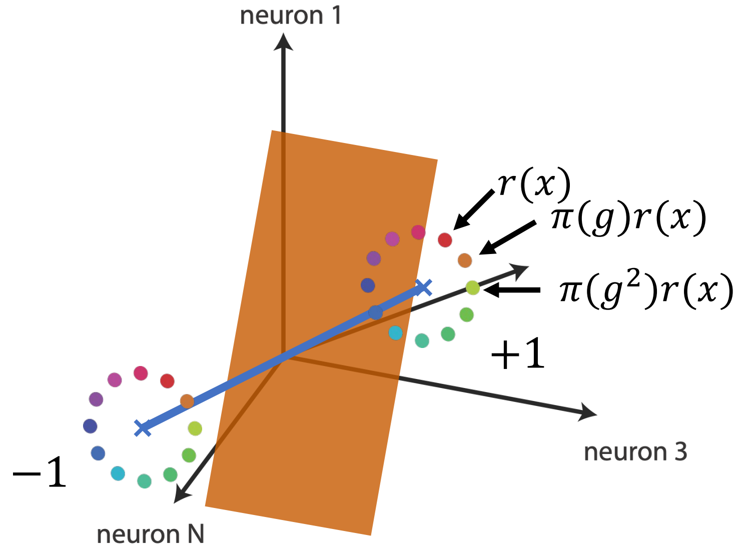


<u>Our main theorem (informal):</u> for P orbits, the fraction of labelings that are linearly separable is $f(P, N_0)$ where

$$N_0 = \mathrm{rank}(\langle \pi(\mathrm{g})\rangle_{\mathrm{g}\in \mathrm{G}})$$

N_0 = dimension of the minimal subspace spanning the centroids of the orbits (blue line)

= number of trivial irreducible representations



Key Lemma

(Informal): Orbits are linearly separable iff their centroids are separable.

Lemma 1. *A dataset* $\{(\pi(g)\boldsymbol{r}^\mu, y^\mu)\}_{g\in G,\mu\in[P]}$ *consisting of* P π*-manifolds with labels* y^μ *is linearly separable if and only if the dataset* $\{(\langle\pi\rangle\boldsymbol{r}^\mu, y^\mu)\}_{\mu\in[P]}$ *consisting of the* P *centroids* $\langle\pi\rangle\boldsymbol{r}^\mu$ *with the same labels is linearly separable. Formally,*

$$\exists \boldsymbol{w}\, \forall g \in G, \mu \in [P] : y^\mu \boldsymbol{w}^\top \pi(g)\boldsymbol{r}^\mu > 0 \iff \exists \boldsymbol{w}\, \forall \mu \in [P] : y^\mu \boldsymbol{w}^\top \langle\pi\rangle\boldsymbol{r}^\mu > 0.$$

Proof of main theorem results from decomposing the representation into irreducible representations and noting that only trivial irrepresentations average to zero due to Schur orthogonality relations.

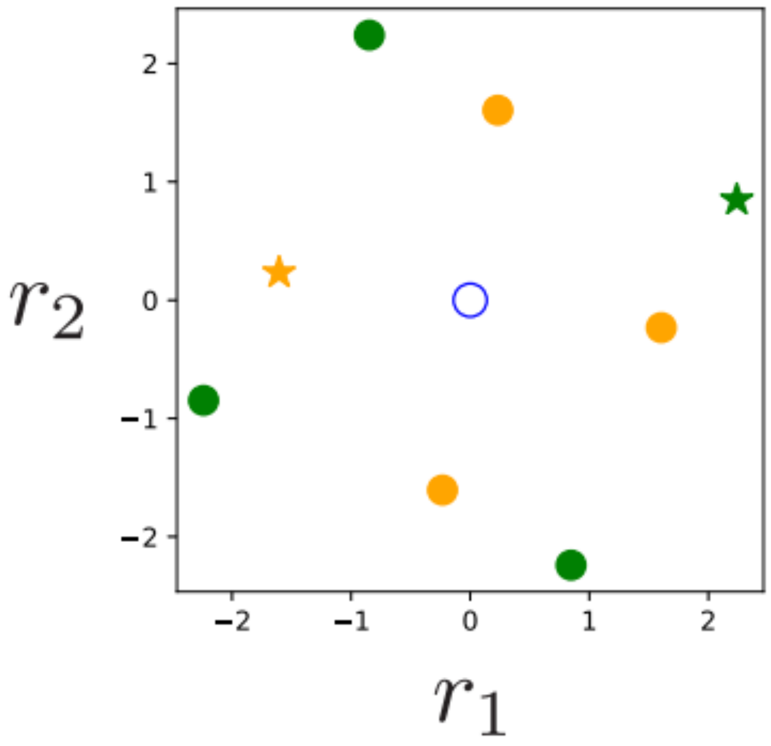
Proof of Key Lemma

Lemma 1. *A dataset* $\{(\pi(g)\boldsymbol{r}^\mu, y^\mu)\}_{g\in G,\mu\in[P]}$ *consisting of* P π*-manifolds with labels* y^μ *is linearly separable if and only if the dataset* $\{(\langle\pi\rangle\boldsymbol{r}^\mu, y^\mu)\}_{\mu\in[P]}$ *consisting of the* P *centroids* $\langle\pi\rangle\boldsymbol{r}^\mu$ *with the same labels is linearly separable. Formally,*

$$\exists \boldsymbol{w}\, \forall g \in G, \mu \in [P] : y^\mu \boldsymbol{w}^\top \pi(g)\boldsymbol{r}^\mu > 0 \iff \exists \boldsymbol{w}\, \forall \mu \in [P] : y^\mu \boldsymbol{w}^\top \langle\pi\rangle\boldsymbol{r}^\mu > 0.$$

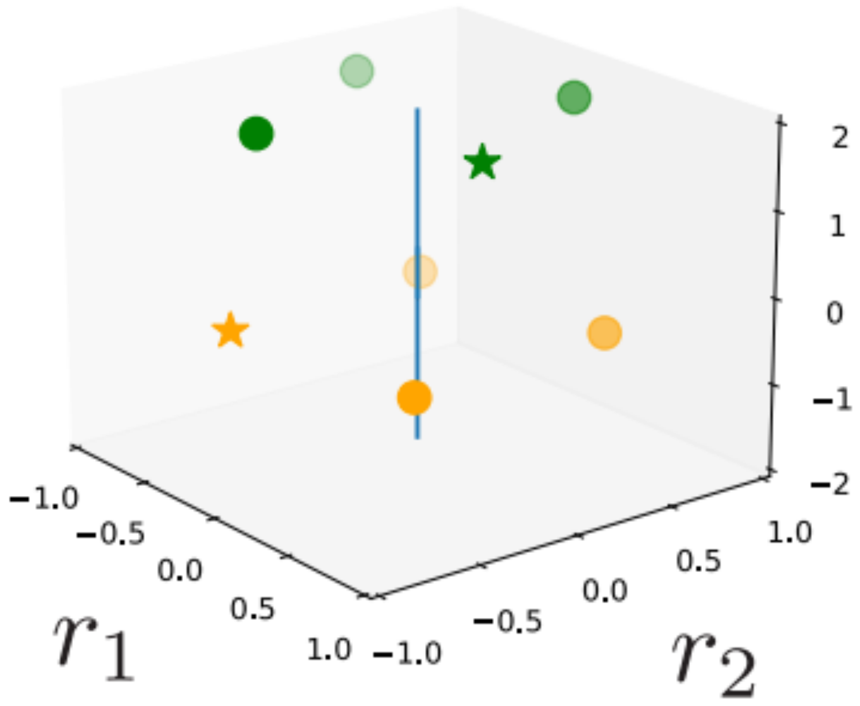
For the reverse implication, suppose $y^\mu \boldsymbol{w}^\top \langle\pi\rangle\boldsymbol{r}^\mu > 0$, and define $\tilde{\boldsymbol{w}} = \langle\pi\rangle^\top \boldsymbol{w}$. We will show that $\tilde{\boldsymbol{w}}$ separates the P π-manfolds since

$$\begin{aligned}
y^\mu \tilde{\boldsymbol{w}}^\top \pi(g)\boldsymbol{r}^\mu &= y^\mu \boldsymbol{w}^\top \langle\pi\rangle\pi(g)\boldsymbol{r}^\mu \quad \text{(Definition of } \tilde{\boldsymbol{w}}\text{)} \\
&= y^\mu \boldsymbol{w}^\top \langle\pi(g')\pi(g)\rangle_{g'\in G}\boldsymbol{r}^\mu \quad \text{(Definition of } \langle\pi\rangle \text{ and linearity of } \pi(g)\text{)} \\
&= y^\mu \boldsymbol{w}^\top \langle\pi\rangle\boldsymbol{r}^\mu \quad \text{(Invariance of the Haar Measure } \mu(Sg) = \mu(S) \text{ for set } S\text{)} \\
&> 0 \quad \text{(Assumption that } \boldsymbol{w} \text{ separates centroids)}
\end{aligned}$$



$N_0 = 0$

$f(2,0) = 0$



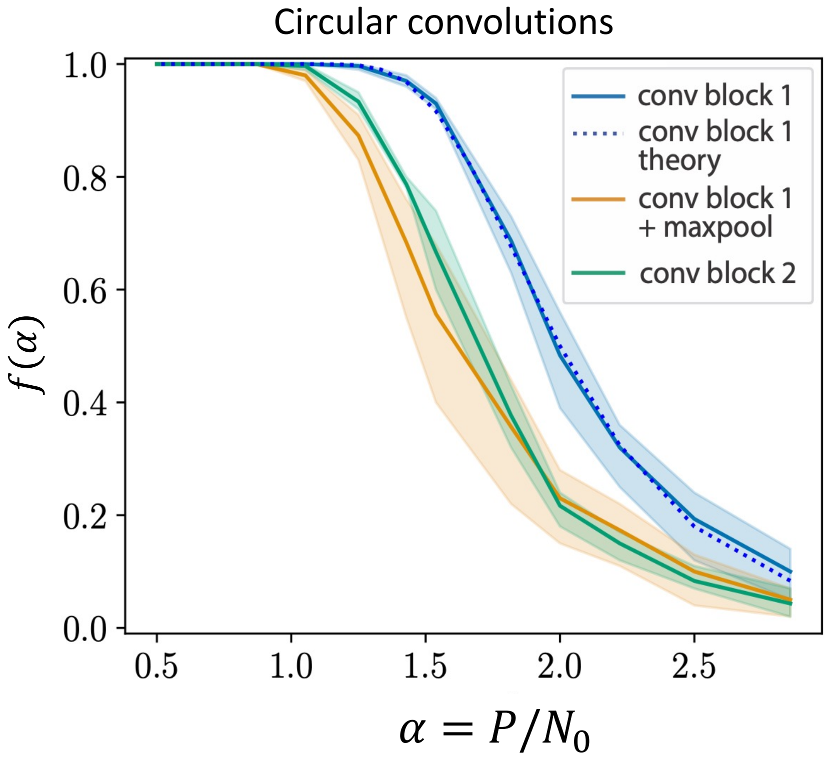
$N_0 = 1$

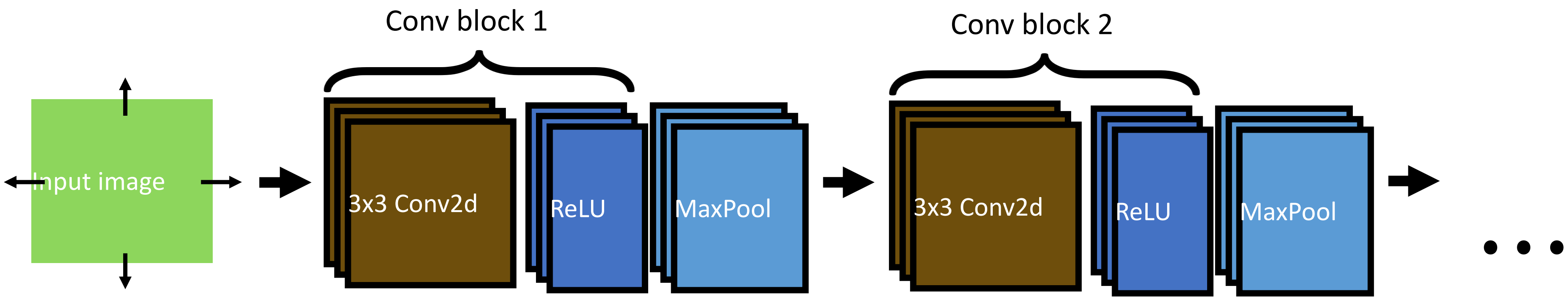
$f(2,1) = 0.5$

Application to Convolutional Neural Networks

- Convolutional neural network
    - A version of VGG-11 trained on CIFAR-10
- For each convolutional layer

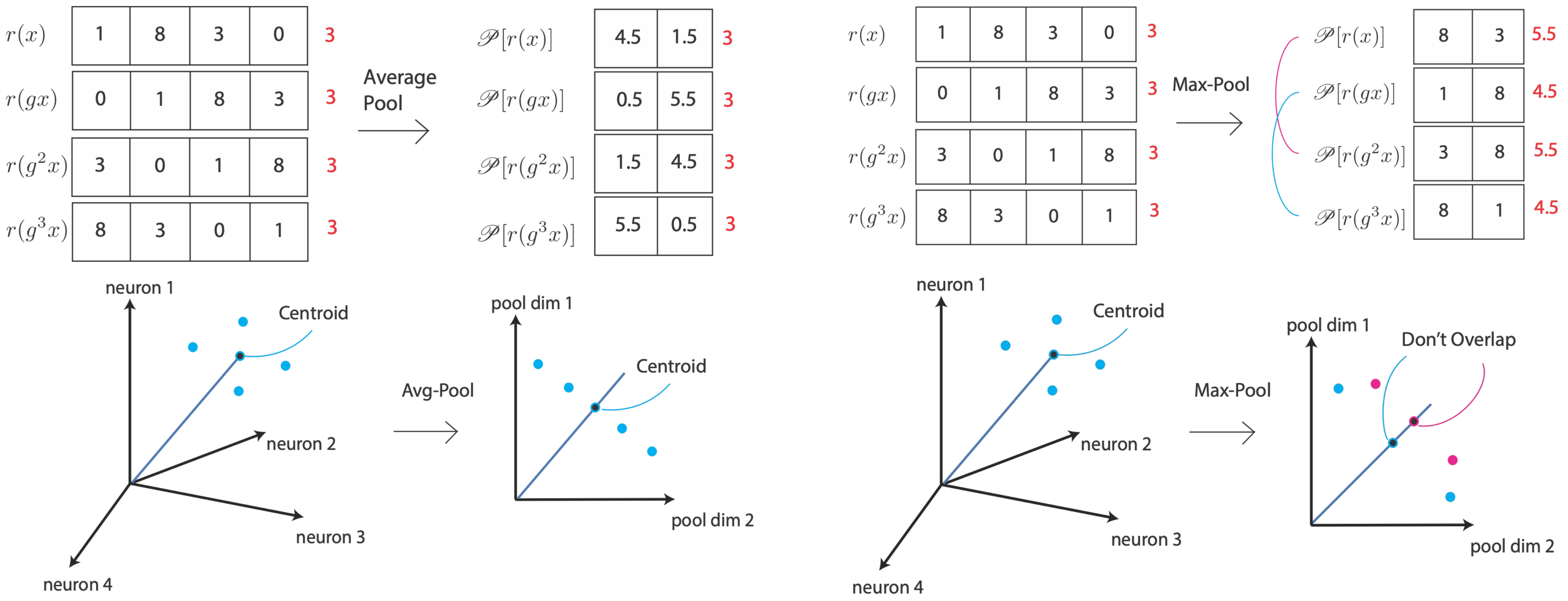
N_0 = #channels used in the layer (proof in paper)





Pooling

- Average pooling: capacity unaffected
- Max pooling: capacity reduced (lower bound in the paper)



Summary

- Neural networks have many trainable parameters
- This work views the capacity of these models from a new perspective by taking into account symmetries
  - Group-invariant perceptron capacity of CNN layers scales with the number of channels in the layer
- Relevant in many situations such as
  - Group-Convolutional neural networks and related models
  - Neural representations in the brain
  - Designing architectures (higher/lower capacity)

Neural networks learn to magnify areas near decision boundaries

Jacob A. Zavatone-Veth JZAVATONEVETH@G.HARVARD.EDU
Department of Physics and Center for Brain Science
Harvard University
Cambridge, MA 02138, USA

Sheng Yang SHENGYANG@G.HARVARD.EDU
John A. Paulson School of Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138, USA

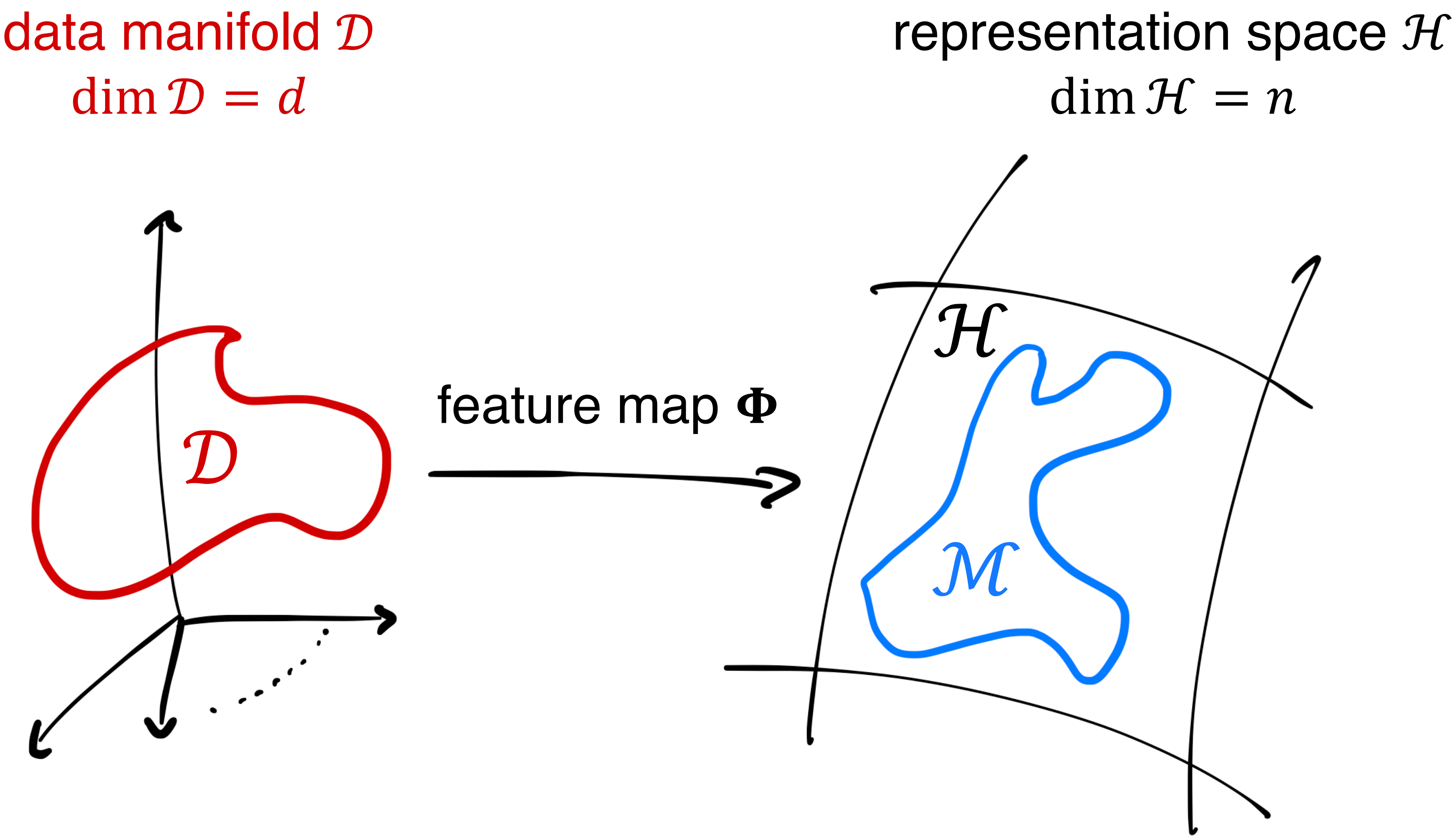
Julian A. Rubinfien JULIAN.RUBINFIEN@YALE.EDU
Department of Physics
Yale University
New Haven, CT 06511, USA

Cengiz Pehlevan CPEHLEVAN@SEAS.HARVARD.EDU
John A. Paulson School of Engineering and Applied Sciences, Center for Brain Science,
and Kempner Institute for Artificial and Natural Intelligence
Harvard University
Cambridge, MA 02138, USA

arXiv:2301.11375

Can we recover useful geometric priors by studying highly performant neural networks?

Setup: Representations as Riemannian Manifolds



representation manifold $\mathcal{M} = \mathbf{\Phi}(\mathcal{D}) \subseteq \mathcal{H}$

Induced Metrics

Given two infinitesimally separated points $\mathbf{x}, \mathbf{x} + d\mathbf{x} \in \mathcal{D}$,

$$ds^2 = \|\mathbf{\Phi}(\mathbf{x} + d\mathbf{x}) - \mathbf{\Phi}(\mathbf{x})\|^2$$

$$= \sum_{i,\mu,\nu} \frac{\partial \Phi_i}{\partial x^\mu} \frac{\partial \Phi_i}{\partial x^\nu} dx^\mu dx^\nu$$

$$= \sum_{\mu,\nu} g_{\mu\nu} dx^\mu dx^\nu$$

If the pullback metric $g_{\mu\nu}$ is positive-definite, $(\mathcal{M}, g)$ is a Riemannian manifold
This opens the door to many analyses, including

- Volume form on $(\mathcal{M}, g)$ is $dV_g = \sqrt{\det g}\, d\mathbf{x}$
- From the metric, we can compute the intrinsic curvature of $(\mathcal{M}, g)$
- We can compute geodesic paths between examples (as in Hénaff & Simoncelli)

Importantly, all these quantities are computable using automatic differentiation, at least in principle

Burges, *Adv. Kernel Methods 1999*; Amari & Wu, *Neural Networks* 1999; Cho & Saul, *arXiv* 2011; Hénaff & Simoncelli, *ICLR* 2016, Hauser & Ray, *NeurIPS* 2017; Benfenati & Marta, *Neural Networks* 2023

Induced Metrics

Given two infinitesimally separated points $\mathbf{x}, \mathbf{x} + d\mathbf{x} \in \mathcal{D}$,

$$ds^2 = \|\mathbf{\Phi}(\mathbf{x} + d\mathbf{x}) - \mathbf{\Phi}(\mathbf{x})\|^2$$

$$= \sum_{i,\mu,\nu} \frac{\partial \Phi_i}{\partial x^\mu} \frac{\partial \Phi_i}{\partial x^\nu} dx^\mu dx^\nu$$

$$= \sum_{\mu,\nu} g_{\mu\nu} dx^\mu dx^\nu$$

$$K(\mathbf{x}, \mathbf{x}') \stackrel{\text{def}}{=} \mathbf{\Phi}(\mathbf{x}) \cdot \mathbf{\Phi}(\mathbf{x}') \qquad \Rightarrow \qquad g_{\mu\nu} = \frac{\partial}{\partial x^\mu} \frac{\partial}{\partial x^\nu} K(\mathbf{x}, \mathbf{x}')|_{\mathbf{x}=\mathbf{x}'}$$

Burges, *Adv. Kernel Methods 1999*; Amari & Wu, *Neural Networks* 1999; Cho & Saul, *arXiv* 2011; Hénaff & Simoncelli, *ICLR* 2016, Hauser & Ray, *NeurIPS* 2017; Benfenati & Marta, *Neural Networks* 2023

PERGAMON

Neural Networks 12 (1999) 783–789

Neural Networks

www.elsevier.com/locate/neunet

Improving support vector machine classifiers by modifying kernel functions

S. Amari*, S. Wu

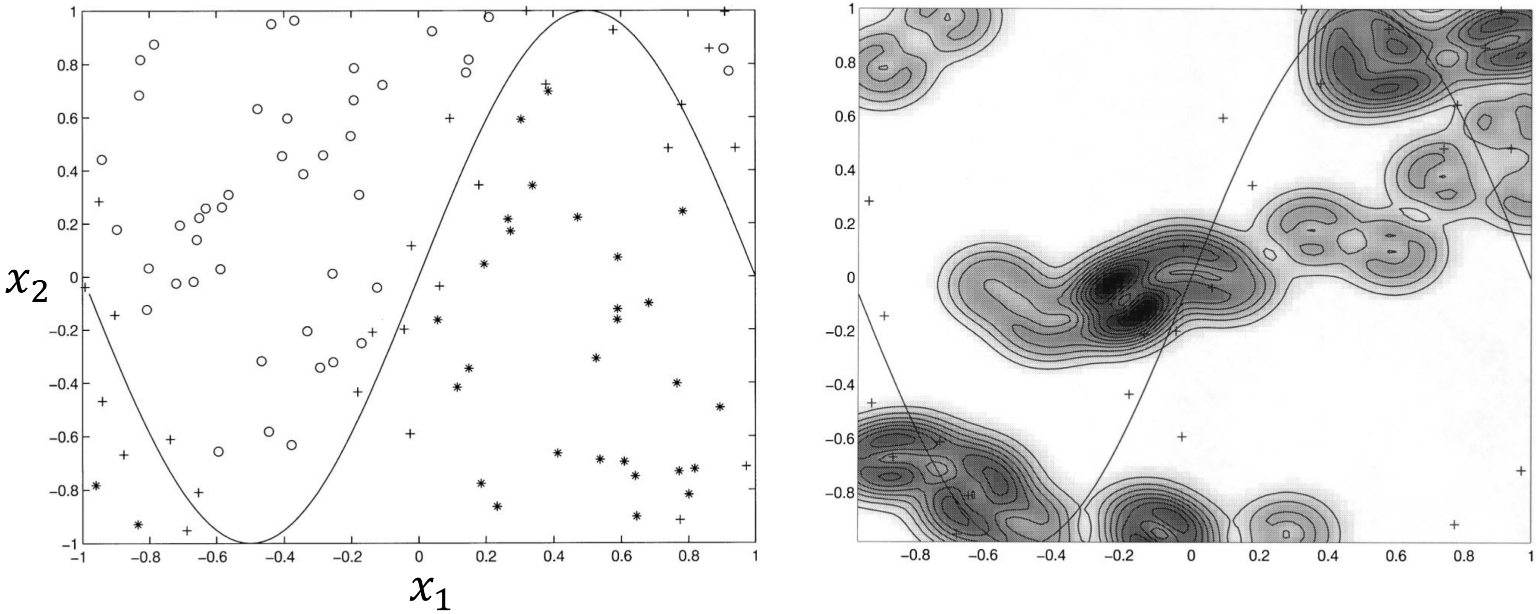
RIKEN Brain Science Institute, The Institute for Physical and Chemical Research, Hirosawa 2-1, Wako-shi, Saitama, Japan

Received 2 February 1999; received in revised form 19 February 1999; accepted 19 February 1999

Idea: larger $\sqrt{\det(g)}$ near decision boundaries → better discriminability of classes

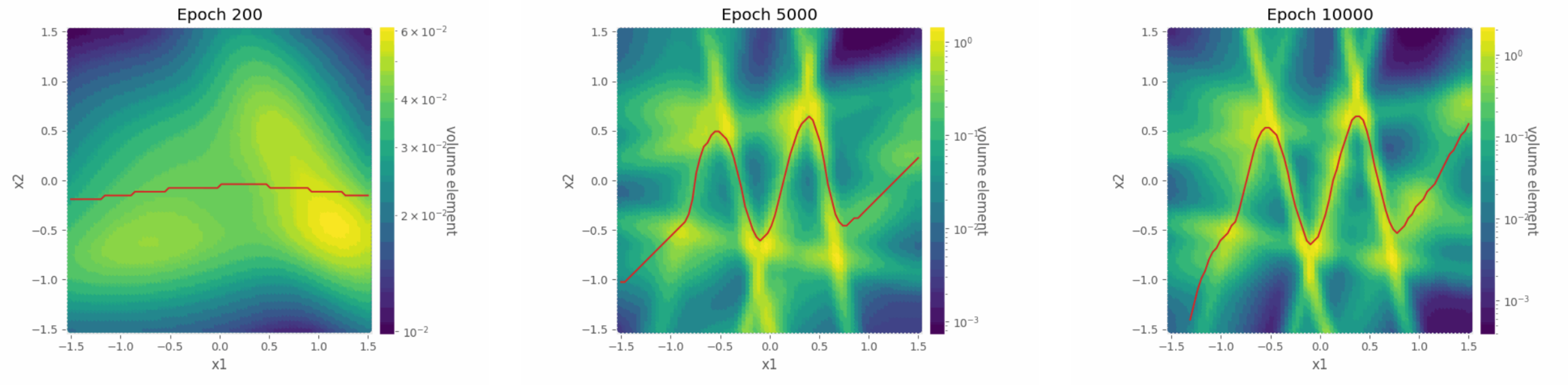
$$k(\mathbf{x},\mathbf{y}) \mapsto h(\mathbf{x})h(\mathbf{y})k(\mathbf{x},\mathbf{y})$$

$$h(\mathbf{x}) = \sum_{\mathbf{v}\in \mathrm{SV}(k)} e^{-\frac{\|\mathbf{x}-\mathbf{v}\|^2}{2\tau^2}}$$



Amari & Wu, *Neural Networks* 1999

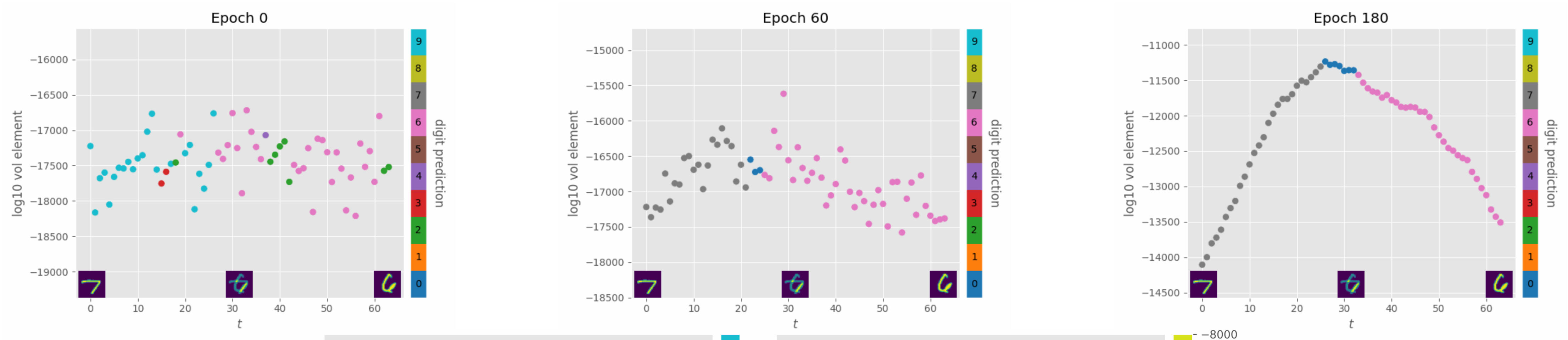
Hypothesis: Deep neural networks trained to perform supervised classification tasks using standard gradient-based methods learn to magnify areas near decision boundaries.

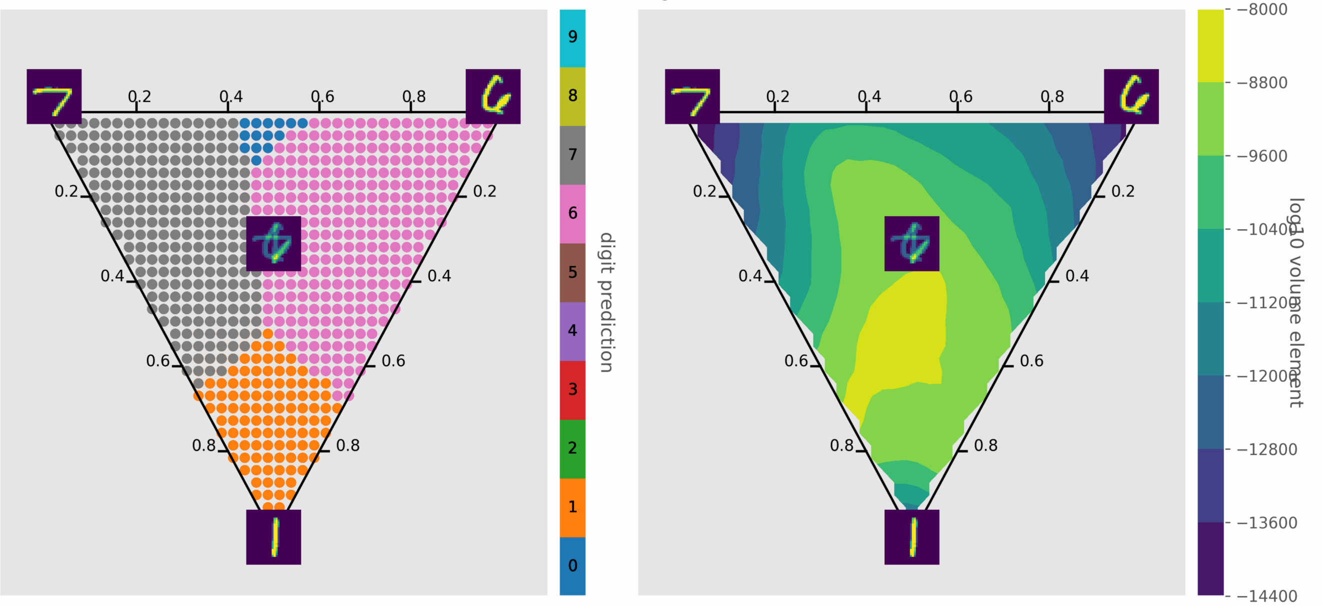


Technical note: this is a MLP with a single hidden layer of 20 sigmoid-activated units.

MLP on MNIST

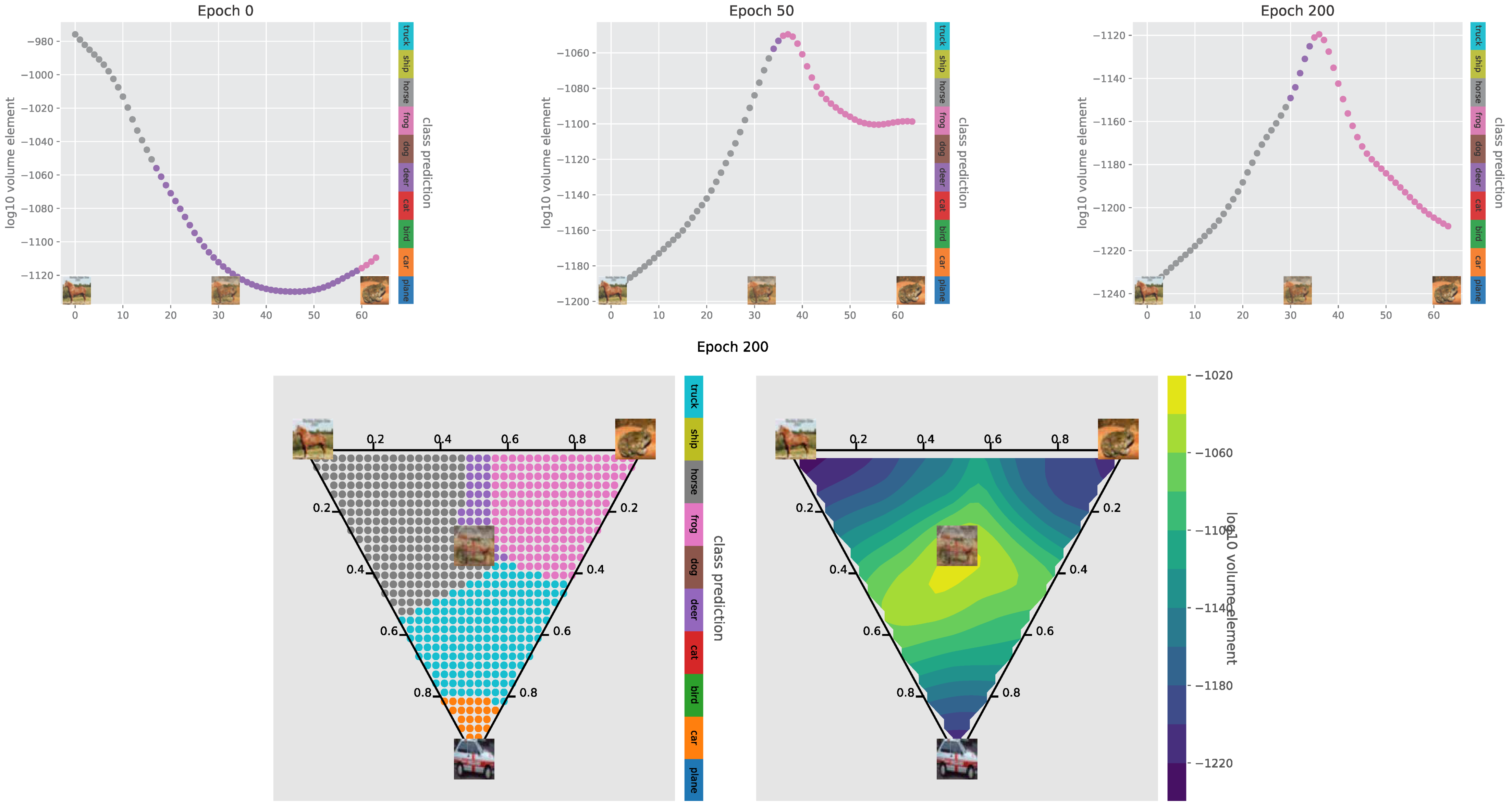
To visualize, interpolate between images in pixel space





Technical note: this is a MLP with a single hidden layer of 2000 sigmoid-activated units.

ResNet-34 on CIFAR-10



Technical note: this network has GELU activations, and we're now using CIFAR-10. Also, this is the last hidden layer's feature map.

Little bit of theory

We can make some progress on understanding these findings in shallow and wide neural networks

Feature map: $\Phi_j(\mathbf{x}) = \frac{1}{\sqrt{n}}\phi(\mathbf{w}_j \cdot \mathbf{x} + b_j)$

Metric: $g_{\mu\nu} = \frac{1}{n}\sum_j \phi'(z_j)^2 w_{j\mu} w_{j\nu}$ $\quad z_j \equiv \mathbf{w}_j \cdot \mathbf{x} + b_j$

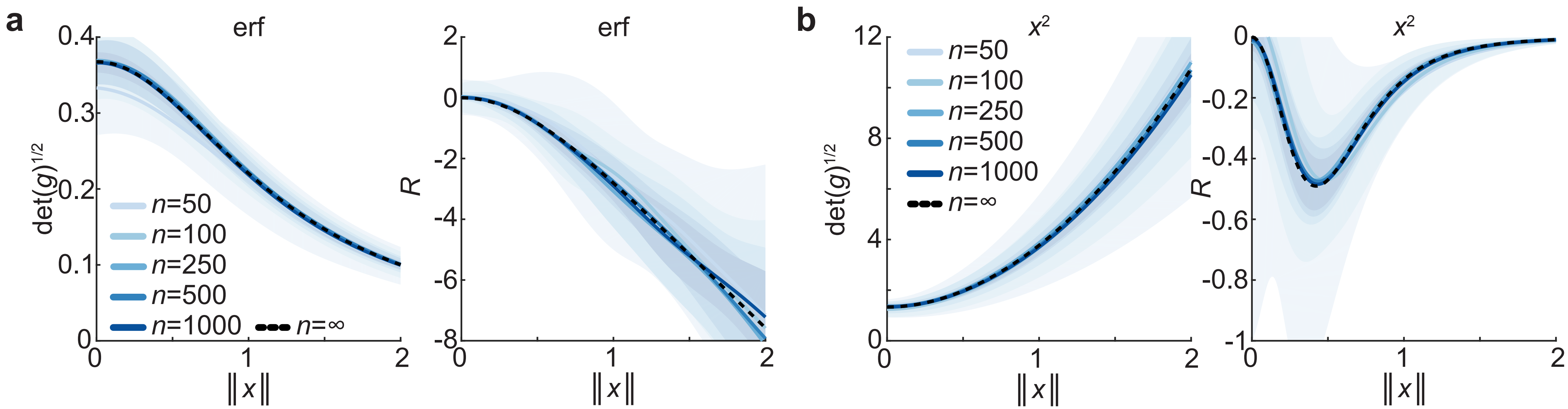
Consider infinite width $n \to \infty$ with $\mathbf{w}_j \sim_{\text{i.i.d.}} \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_d)$, $\quad b_j \sim_{\text{i.i.d.}} \mathcal{N}(0, \zeta^2)$, then at initialization:

$$g_{\mu\nu} = e^{\Omega(\|\mathbf{x}\|^2)}[\delta_{\mu\nu} + 2\Omega'(\|\mathbf{x}\|^2) x_\mu x_\nu]$$

$$e^{\Omega(\|\mathbf{x}\|^2)} = \sigma^2 \mathbb{E}[\phi'(z)^2], \quad z \sim \mathcal{N}(0, \sigma^2\|\mathbf{x}\|^2 + \zeta^2)$$

$$\det g = e^{\Omega d}(1 + 2\|\mathbf{x}\|^2\Omega')$$

$$R = -\frac{3(d-1)e^{-\Omega}(\Omega')^2\|\mathbf{x}\|^2}{(1+2\|\mathbf{x}\|^2\Omega')^2}\left[d + 2 + 2\|\mathbf{x}\|^2\left((d-2)\Omega' + 2\frac{\Omega''}{\Omega'}\right)\right]$$


a
erf
erf
det(g)^1/2
R
n=50
n=100
n=250
n=500
n=1000
n=∞
‖x‖
b
x²
x²
0
0.4
0.3
0.2
0.1
2
-2
-4
-6
-8
12
8
4
-0.2
-0.4
-0.6
-0.8
-1
1

Little bit of theory

To see how the geometry of the pullback metric changes during training, consider a Bayesian neural network, i.e. we fix isotropic standard Gaussian priors over weight, and choose likelihood

$$p(\{(\mathbf{x}_a, \mathbf{y}_a)\}_{a=1}^{p} \mid \mathbf{W}, \mathbf{V}) \propto \exp\left(-\frac{\beta}{2}\sum_{a=1}^{p}\left\|\frac{1}{\sqrt{n}}\sum_{i=1}^{n}\mathbf{v}_i\phi(\mathbf{w}_i \cdot \mathbf{x}) - \mathbf{y}_a\right\|_2^2\right)$$

For a quadratic activation function and interpolating a single training example $(\mathbf{x}_a, \mathbf{y}_a)$

$$\frac{\langle\sqrt{\det g}\rangle}{\sqrt{\det \bar{g}}} = 1 + \frac{1}{n}\left(\frac{y_a^2}{\|\mathbf{x}_a\|^4} - 1\right)\left(\frac{4}{3}\rho^4 + (d+2)\rho^2 + 1\right) + \mathcal{O}\left(\frac{1}{n^2}\right)$$

$$\rho = \mathbf{x}_a \cdot \mathbf{x}/(\|\mathbf{x}_a\|\|\mathbf{x}\|)$$

Do deep networks learn to magnify areas near decision boundaries?

Yes!

Acknowledgments

Alex Atanasov
Blake Bordelon
Hamza Chaudhry
Ganesh Kumar
Adam Lee
Mary Letey
Paul Masset
Sab Sainathan
Shanshan Qin
Ben Ruben
William Tong
Nikhil Vyas
Ningjing Xia
Sheng Yang
Jacob Zavatone-Veth

<u>Collaborators</u>
Shubhendu Trivedi
Julian Rubinfien
Matthew Farrell






Harvard John A. Paulson School of Engineering and Applied Sciences

Kempner Institute
For the Study of Natural & Artificial Intelligence at Harvard University

HDSI
intel
Google
NIH
NSF
ALFRED P. SLOAN FOUNDATION