# **InPCA as a lense**

• Mao, Griniasty, Yang, Teoh, Transtrum, Sethna & Chaudhari. *arXiv:2305.01604*.

**The Training Process of Many Deep Networks Explores the Same Low-Dimensional Manifold**

**Itay Griniasty griniasty@cornell.edu**



# DNNs the canonical synthetic emergent machine



Me: Hi, please create an image representing the complex functionalities large language models can perform.

ChatGPT4: Here is the image representing the complex functionalities of large language models. It depicts a futuristic and intricate machine symbolizing a large language model, surrounded by diverse groups of people engaging with it. The scene captures the advanced technology and intelligence of these models.

## MANY parameters MANY outputs



# A Sketch of a Neural Network





### Hidden Layers

# A Sketch of Training

Input





## Minimize **Loss** = Difference between Prediction and Truth

This is a high-dimensional ( weights), large scale (millions of images) and non-convex optimization problem  $10^6 - 10^{12}$ 

How can we train a  $10^{11}$  dimensional machine in a year? 1011

How Much Do The Architecture / Algorithm / Augmentation Matter?

### Why Can We Train DNNs? NB Neural network training relies on our ability to find "good" minimizers of highly non-convex loss functions. It is well-known that certain network architecture

chosen training parameters (batch size, learning rate, optimizer) produce minimiz-

(a) with skip connections (b) with skip connections (b) with skip connections (b) with skip connections (b) with skip connections ( Loss Landscape Li et al. NeurIPS 2018



# Many DNN Designs, What Is The Difference?







Transformer Network

Recurrent Network

- 
- 





### Residual Network



# Many DNN Designs, What Is The Difference?





Transformer Network

Recurrent Network

AlexNet



# How Do We Visualize the Output?

Network Represented over N Samples

Sample *x* Network with Weights *w* 





ImageNet:  $10^6$  Images, 1,000 Classes  $\sim 10^9$  Dimensions  $10^6$  Images, 1,000 Classes ~  $10^9$ 

## Dog

### CIFAR-10:  $5 \cdot 10^4$  Images, 10 classes  $\sim 5 \cdot 10^5$  Dimensions  $5 \cdot 10^4$  Images, 10 classes  $\sim 5 \cdot 10^5$

s: 
$$
P_w(\vec{y}) = \prod_{n=1}^{N} p_w^n(y_n | x_n)
$$
  
Class y

8

all probabilities are orthogonal

Even if 
$$
p_1 \cdot p_2 = 1 - \epsilon
$$
  
For many samples  $P = \prod_{n=1}^{N} p^n$ 

Natural metric can't discern close from far

1  $\cdot p_2^n \approx 1 - e^{-N\epsilon}$  $\sim 1 - \rho^{-N}$  $1$ 

$$
P_1 \cdot P_2 = \prod_n p_1^n \cdot p_2^n \approx 0
$$

$$
d_H^2(P_1, P_2) = 0.5 ||P_1 - P_2||^2 = 1 - \prod p_1^n
$$





*n*



### High Dimensional Output is Cursed We are now poised to define the intensive distance by taking the number of replicas to zero:



### How Do We Visualize High Dimensional Data? **Fig. 2.** Replicated Ising model illustrating the derivation of our inten-

h✓**1**; ✓**2**i

sive embedding. All points are colored by magnetic field strength. (*A*)

*L* ( *{x***1**, *...* , *xN} |* ✓))

(*<sup>N</sup>* ) <sup>=</sup> *<sup>L</sup>* (*x***<sup>1</sup>** *<sup>|</sup>* ✓)*···L* (*xN <sup>|</sup>* ✓), **[6]**

Because of this, the intensive embedding can overcome the loss

of relative contrast (19) discussed at the beginning of this section.

Distances in the intensive embedding maintain distinguishabil-

ity in high dimensions, as illustrated in Fig. 2*B*, wherein the

 $2D$  nature of the Ising model has been recovered. We hypothesis been recovered. We have  $\mathcal{L}_\mathcal{D}$ 

esize that this process, which curse of dimensionality of dimensionality of  $\mathcal{O}_\mathbf{C}$ 

for models with too many samples, with too many samples, will also consider  $\mathcal{C}$ 

els with intrinsically high dimensionality. The intensive distance



### inherent 2D structure of the Ising model has been recovered.  $T \cap \bigcap \bigcap \{I\cap A | I\cap A\}$ tance between replicated models. The likelihood for *N* replicas **Intensive Embeddings**  $\blacksquare$ sphere, just like the larger, 4  $\blacksquare$ systems have "too much information," in the same way that large numbers of samples have too much information. In the



where the normalization constraint of *L* (*x |* ✓)forces *z<sup>x</sup>* to lie on



Quinn et al. PNAS 2019  $\alpha$ ullillicial. Throe colored by magnetic field by magnetic  $\alpha$ 

# Computational Intensive Geometry





 $\textsf{Truth: } P_* = \delta_{\vec{\textnormal{y}}*}(\vec{\textnormal{y}}), \vec{\textnormal{y}}* = \textsf{True}$  label

## The Training Process explores a Low Dimensional Manifold



## ~2,000 Configurations

## ~150,000 Networks

- **Architectures**
- Optimization:
	- SGD, SGDN, ADAM
- **Hyper Parameters** 
	- Learning Rate, Batch Size
- **Regularization**
- Data augmentation
- 10 Random seeds



## The Training Process explores a Low Dimensional Manifold



## ~2,000 Configurations

## ~150,000 Networks

- **Architectures**
- Optimization:
	- SGD, SGDN, ADAM
- Hyper Parameters
	- Learning Rate, Batch Size
- **Regularization**
- Data augmentation
- 10 Random seeds



# Intensive Embedding are Minkowski ℝ*q*,*p*−*<sup>q</sup>*

### **Light Cones Connects Different Models With Equal Predictions**  $dx = \pm dt$



$$
ds^2 = dx^2 - dt^2
$$







 $\eta$  $t<sup>2</sup>$ 

## Intensive Embedding are Minkowski ℝ*q*,*p*−*<sup>q</sup>* HAN KHENG TEOH *et al.* PHYSICAL REVIEW RESEARCH **2**, 033221 (2020)



Teoh et al. PRR 2020 Model Manifold of the 2D Ising model **Connects** I

### phase transition. The Ising model manifold is embedded into (2 + 2)  $\mathbf{v} - \boldsymbol{+} d\boldsymbol{t}$  $\mathcal{W} = \mathcal{W}$ erent iviodels **With Equal Predictions Light Cones Connects Different Models**  $dx = \pm dt$



$$
ds^2 = dx^2 - dt^2
$$

# Test Embedding is also Low Dimensional



# What Can We Learn About Different Configurations?

- Architectures
- Optimization:
	- SGD, SGDN, ADAM
- Hyper Parameters
	- Learning Rate, Batch Size
- Regularization
- Data augmentation
- 10 Random seeds





## ~2,000 Configurations ~150,000 Networks

## A Larger Network Trains Along the Same Manifold as a Smaller Network With a Similar Architecture (But is Faster)







# Distance between trajectories



### Define progress  $\approx$  "geodesic arclength parametrization" to remove speed  $d(\tau_u, \tau_v) = \int d_B(P_{u(s)}, P_{v(s)}) ds$ 0.23 Fully. **Connected** InPC1**Geodesic**  $0.0$ Geodesic **Progress Small Large**  $S_W$ Trajectory  $-0.13$ ResNet  $0.0$ <sup>ResNet</sup> $0.16$ InPC2



## Architecture- Not training or regularization - primarily distinguished trajectories





# Why is the training low dimensional?

## **Suspects**



- 1. Data is Structured Easy and hard Images are common across networks.
- 2. Weights Initialize at ignorance  $P_0$
- 3. Data is Low Dimensional



# Why Are The Training Manifolds Low Dimensional?

## **Suspects**



- 1. Data is Structured Easy and hard Images are common across networks.
- 2. Weights Initialize at ignorance  $P_0$
- 3. Data is Low Dimensional

### **Experiments**

- 1. Embed Data using Initial and Final **Tangents**
- 2. Initialization in multiple corners
- 3. Train on synthetic data with varying effective dimensionality & initialization

# Experiment 1: Embed Data using Initial and Final Tangents





Original InPCA Tangent Embedding Explained Pairwise Distance

InPCA using 4 points

# Experiment 2: Initialization in multiple corners



### Training Data **Test Data**





# Experiment 3: Synthetic data with varying dimensionality





## Experiment 3.2: Synthetic data with varying dimensionality & initialization





**SLOPPY INPUTS** 

### **NON-SLOPPY INPUTS**



## Why Are The Training Manifolds Low Dimensional? a hint to why training neural nets is easy, and why they generalize well

- 1. Data is Structured Hard/ Easy images are common
- 2. Data is Sloppy
- 3. Weights Initialized at hypothesis



## Summary: Intensive Embeddings Uncover that Neural Networks Learn in The Same Way



- 1. InPCA  $\Rightarrow$  Computationally feasible distance in high D.
- 2. Sloppiness  $\Rightarrow$  Low D visualization
- 3. The Training of Neural Networks Explores the Same Low D Manifold
- 4. Configuration distance ⇒ Variation in the path of learning is mostly due to architecture, not optimization technique



## The Manifold of Typical Learnable Tasks is Also Low Dimensional

Ramesh et al. ICML 2023

