Understanding Deep Learning

Yasaman Bahri
Google Research

IAIFI Colloquium
Nov 2021

Theoretical understanding of deep learning

Can we gain an understanding of deep learning?

- One that gives us insight into *mechanisms* and matches with experiments
- A different set of demands than in learning theory (statistical / algorithmic guarantees)
- Some questions may be too microscopic (byproduct of a complex optimization process)

Part 1: Highlights (from past several years) on theory of learning in deep neural networks (DNNs)

- Exactly solvable limits from in DNNs that are wide
    - Correspondence with Gaussian processes, linear models, & kernel methods
- Dynamics in other regimes

Part 2: Understanding "scaling laws" in (supervised) deep learning

Class of functions

Fully-connected deep neural network (DNN)

- Input $x \in \mathbb{R}^d$
- Each layer ℓ has parameters (weights, biases) W^ℓ_{ij}, b^ℓ_i
- Hidden layer width: n
- Nonlinearity $\phi(\cdot)$
- Parameters (collectively): $\{\theta_\mu\}$

$$f^0_i(x) = b^0_i + \sum_{j=1}^{d} W^0_{ij} x_j$$

$$\downarrow$$

$$f^1_i(x) = b^1_i + \sum_{j=1}^{n} W^1_{ij} \phi(f^0_j(x))$$

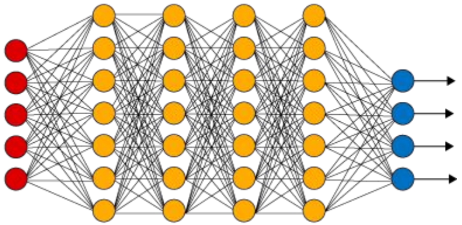
$$\downarrow$$

$$\vdots$$

$$\downarrow$$

$$f^\ell_i(x) = b^\ell_i + \sum_{j=1}^{n} W^\ell_{ij} \phi(f^{\ell-1}_j(x))$$

Setting of interest: empirical risk minimization under gradient descent



$$\mathcal{L}(\theta) = \frac{1}{D} \sum_{\alpha=1}^{D} \ell(f_\theta(x^\alpha), y^\alpha)$$

$$\Delta\theta_\mu = -\eta \frac{\partial \mathcal{L}(\theta)}{\partial \theta_\mu}$$

Function Space Description

Historically, a lot of focus has been on parameter space because:

- That is what you access directly in optimization.
- Statistical models were constructed as to be more interpretable: learned parameters had meaning.
    - e.g. moments of a distribution

In neural networks, parameters were not designed to be interpretable, and there is redundancy.

Motivates asking questions about the *functions that are represented*.

What class of functions is expressed at initialization (beginning of optimization)?

Infinite-width limits: an exact equivalence to Gaussian field theory

We typically draw parameters iid from a prior distribution.

$$p(W_{ij}^{\ell}) \;\sim\; \mathcal{N}(0, \sigma_w^2/n),\;\; p(b_i^{\ell}) \;\sim\; \mathcal{N}(0, \sigma_b^2)$$

$$\boxed{f_i^{\ell}(x) = b_i^{\ell} + \sum_{j=1}^{n} W_{ij}^{\ell}\phi(f_j^{\ell-1}(x))}$$

Main idea: as $n \to \infty$, we can apply the Central Limit Theorem to get that any collection $\{f_i^{\ell}(x^{\alpha}), f_i^{\ell}(x^{\beta}), ...\}$ will be Gaussian distributed.

The distribution over functions is described by a new class of Gaussian Processes:

$$f_i^{\ell} \sim \mathcal{GP}(0, K^{\ell})$$

$$\begin{aligned} K^{\ell}(x, x') &= \mathbb{E}\left[f_i^{\ell}(x) f_i^{\ell}(x')\right] \\ &= \sigma_b^2 + \sigma_w^2\, \mathbb{E}_{f^{\ell-1} \sim \mathcal{GP}(0, K^{\ell-1})}\left[\phi(f_i^{\ell-1}(x))\phi(f_i^{\ell-1}(x'))\right] \end{aligned}$$

"NNGP" kernel

$$\boxed{K^{\ell}(x, x') = \sigma_b^2 + \sigma_w^2\, \mathcal{C}_{\phi}\Big(K^{\ell-1}(x, x'), K^{\ell-1}(x, x), K^{\ell-1}(x', x')\Big)}$$

Lee* and YB*, et al. ICLR 2018; A. G. de G. Matthews, et al. ICLR 2018.

Infinite-width limits: an exact equivalence to Gaussian field theory

Recursion on function → recursion on kernels

$$\begin{tikzcd}
f_i^0(x) = b_i^0 + \sum_{j=1}^{d} W_{ij}^0 x_j \arrow[d] & & K^0 \arrow[d, "\mathcal{C}"] \\
f_i^1(x) = b_i^1 + \sum_{j=1}^{n} W_{ij}^1 \phi(f_j^0(x)) \arrow[d] \arrow[rr, "n \to \infty"] & & K^1 = \sigma_b^2 + \sigma_w^2 \, \mathbb{E}_{(f,f')\sim\mathcal{N}(0,K^0)} \left[\phi(f)\phi(f')\right] \arrow[d, "\mathcal{C}"] \\
\vdots \arrow[d] & & \vdots \arrow[d, "\mathcal{C}"] \\
f_i^\ell(x) = b_i^\ell + \sum_{j=1}^{n} W_{ij}^\ell \phi(f_j^{\ell-1}(x)) \arrow[rr, "n \to \infty"] & & K^\ell = \sigma_b^2 + \sigma_w^2 \, \mathbb{E}_{(f,f')\sim\mathcal{N}(0,K^{\ell-1})} \left[\phi(f)\phi(f')\right]
\end{tikzcd}$$

$$K^0(x, x') = \mathbb{E}\left[f_i^0(x) f_i^0(x')\right] = \sigma_b^2 + \sigma_w^2 \left(\frac{x \cdot x'}{d}\right)$$ Base case

Dynamics of gradient descent in infinitely wide DNNs

Consider gradient flow (for illustration):

$$\frac{d\theta_\mu}{dt} = -\frac{\partial \mathcal{L}}{\partial \theta_\mu} = -\sum_{\alpha \in \mathcal{D}} \frac{\partial \mathcal{L}}{\partial f(x^\alpha)} \frac{\partial f(x^\alpha)}{\partial \theta_\mu}$$

$$\left(\text{Could consider } \frac{\partial \mathcal{L}}{\partial f(x^\alpha)} = f(x^\alpha) - y^\alpha \right)$$

Study function space evolution:

$$\frac{df(x)}{dt} = \sum_\mu \frac{\partial f(x)}{\partial \theta_\mu} \frac{\partial \theta_\mu}{\partial t} = -\sum_{\alpha \in \mathcal{D}} \frac{\partial \mathcal{L}}{\partial f(x^\alpha)} \left(\sum_\mu \frac{\partial f(x)}{\partial \theta_\mu} \frac{\partial f(x^\alpha)}{\partial \theta_\mu} \right)$$

This suggests defining a quantity, which is a kernel ("Neural Tangent Kernel").
It holds a distinguished place in the dynamics.

$$\boxed{\Theta_t(x, x') \equiv \sum_\mu \frac{\partial f(x)}{\partial \theta_\mu} \frac{\partial f(x')}{\partial \theta_\mu}}$$

Dynamics of gradient descent in infinitely wide DNNs

$$\text{Amazingly, as } n \to \infty, \ \frac{d\Theta_t(x, x')}{dt} = 0 \text{ and so } \Theta_t(x, x') = \Theta_0(x, x').$$

This allows us to get exact, analytic solutions for the evolution.

Furthermore, it turns out that the "effective" model in parameter space corresponds to a first-order Taylor expansion of the function:

$$\boxed{f_t(x) = f_0(x) + \sum_{\mu} \frac{\partial f_0(x)}{\partial \theta_\mu} \cdot \Big(\theta_\mu(t) - \theta_\mu(0)\Big)}$$

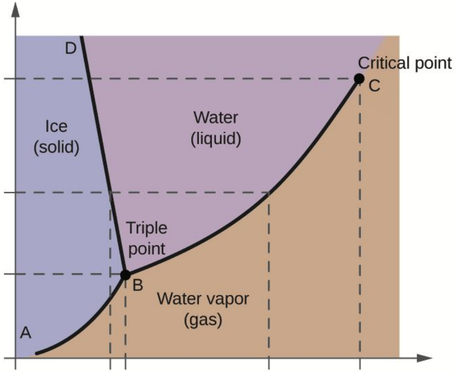
Mechanism by which this happens: crucially tied to how various scales enter the problem.

[1]. A. Jacot, et al. "Neural Tangent Kernel," NeurIPS 2018, and many others. **(Function space evolution.)**
[2]. Lee*, Xiao*, Schoenholz, YB, Novak, Sohl-Dickstein, Pennington. NeurIPS 2019. **(Parameter space evolution.)**

Feature learning regimes

NTK - GP correspondence breaks down above a critical learning rate [1] -> phase transition.

Perturbation theory for corrections based off of GP limit & other theoretical treatments (see [2-6]).



Depth of network
Width of hidden layers
Dataset size
Learning rate
….
play a role in determining which part of the "phase diagram" you are in.

[1]. Lewkowycz, YB, Dyer, Sohl-Dickstein, Gur-Ari, arxiv 2003.02218.
[2]. Yaida, MSML 2020.
[3]. Gur-Ari & Dyer, ICLR 2020.
[4]. Halverson, Maiti, Stoner, MLST, Vol 2 035002 (2021).
[5]. Roberts, Yaida, Hanin, arxiv 2106.10165 (2021).
[6] Zavatone-Veth & Pehelvan, NeurIPS 2021 & subsequent papers.

Lessons learned from the infinite width limit

Infinitely many parameters is not a problem.

- Less complex as we add parameters and approach the limit.

Reduced a complicated problem to a simpler problem: linear models with special set of fixed features.

(Not discussed) From the connection to linear models, know that a much smaller number of directions in parameter space are updated (determined by data).

(Not discussed) From the exact solutions, one can see that “simpler” functions are learned first during training (“spectral bias”).

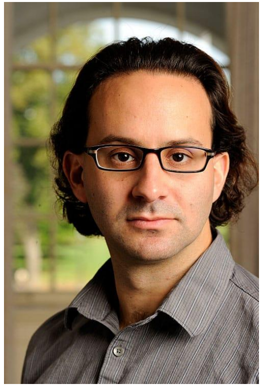
These are statements about the *effective dynamics*, i.e. the region actually explored.

The problem turned out to be effectively convex.

Understanding Neural Scaling Laws

Ethan Dyer
(Google)

Jaehoon Lee
(Google)



Jared Kaplan
(JHU)

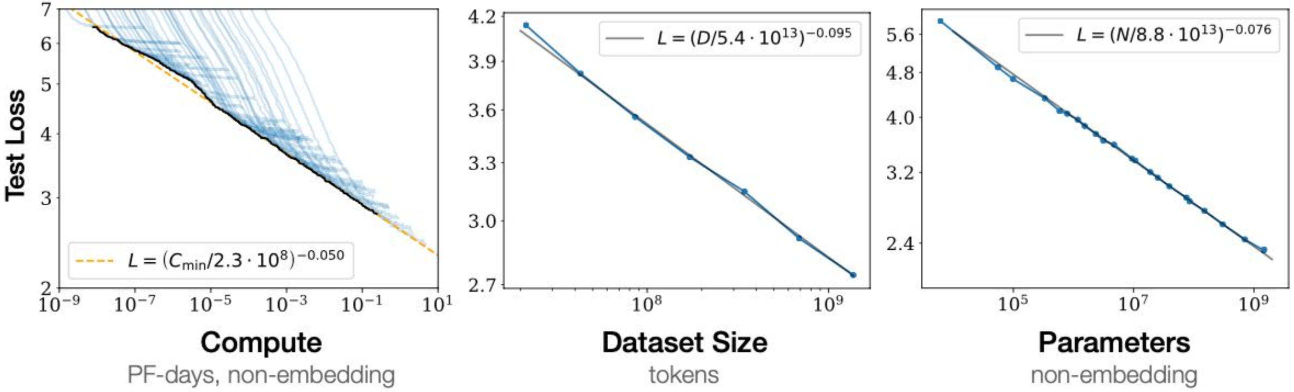
Utkarsh Sharma
(JHU)

Motivation

Recent empirical work has found that in many settings in practice, neural network performance obeys smooth power-law trends as

- Dataset size
- Model size
- Amount of compute

is increased.



Kaplan & McCandlish, et al. “Scaling Laws for Neural Language Models.” arxiv 2001.08361.

Decoder-only Transformer model trained on WebText2 dataset.

Motivation

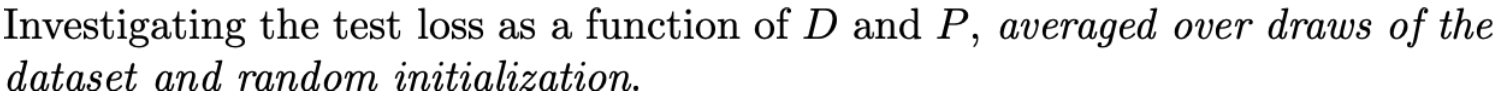
Can we better understand why these trends emerge empirically, and what features of the data and models determine the exponents?

Understand the mechanisms that control scaling

- We will see in some regimes there is a universal exponent and in others it is non-universal.
- We will examine a setting in which we have a full handle on the problem (random feature / kernel models).
- In the more general setting, we will propose an expansion & will empirically test the connection to other measurable quantities.

Setup

Let D = training set size for dataset $\mathcal{D}$
Let P = number of models parameters; **mostly we will focus on just scaling with layer width, W ~ P$^{1/2}$**.



Investigating the test loss as a function of D and P, *averaged over draws of the dataset and random initialization.*

$$L = \mathbb{E}_{\mathcal{D},\theta_0}\, \mathbb{E}_{(x,y)\sim\mathcal{P}} \left[\mathcal{L}(\hat{f}_{\mathcal{D},\theta_0}(x), y)\right]$$

Notation

Power law exponent for D scaling: α_D
Power law exponent for P scaling: α_P

$$L(D) = \frac{C_0}{D^{\alpha_D}} + C_\infty + ...$$

$$L(P) = \frac{C'_0}{P^{\alpha_P}} + C'_\infty + ...$$

Classification

We devised a classification of exponents based on their origin.

“Variance-limited” regime:

- Originates from fluctuations (the variance) when *smoothly* approaching a limit.
- Gives universal integer exponent = 1 asymptotically.

“Resolution-limited” regime:

- Main idea: when one variable is not a bottleneck (take it to -> infinity), study scaling as a function of the other variable. This variable (training data or parameters) serves to improve the resolution of some manifold.
- Gives nontrivial (non-integer) exponents.

Classification

Variance-limited regime: Fix one of D or P. Let other variable grow (>>) and study scaling with that variable.
Integer (=1) exponent.

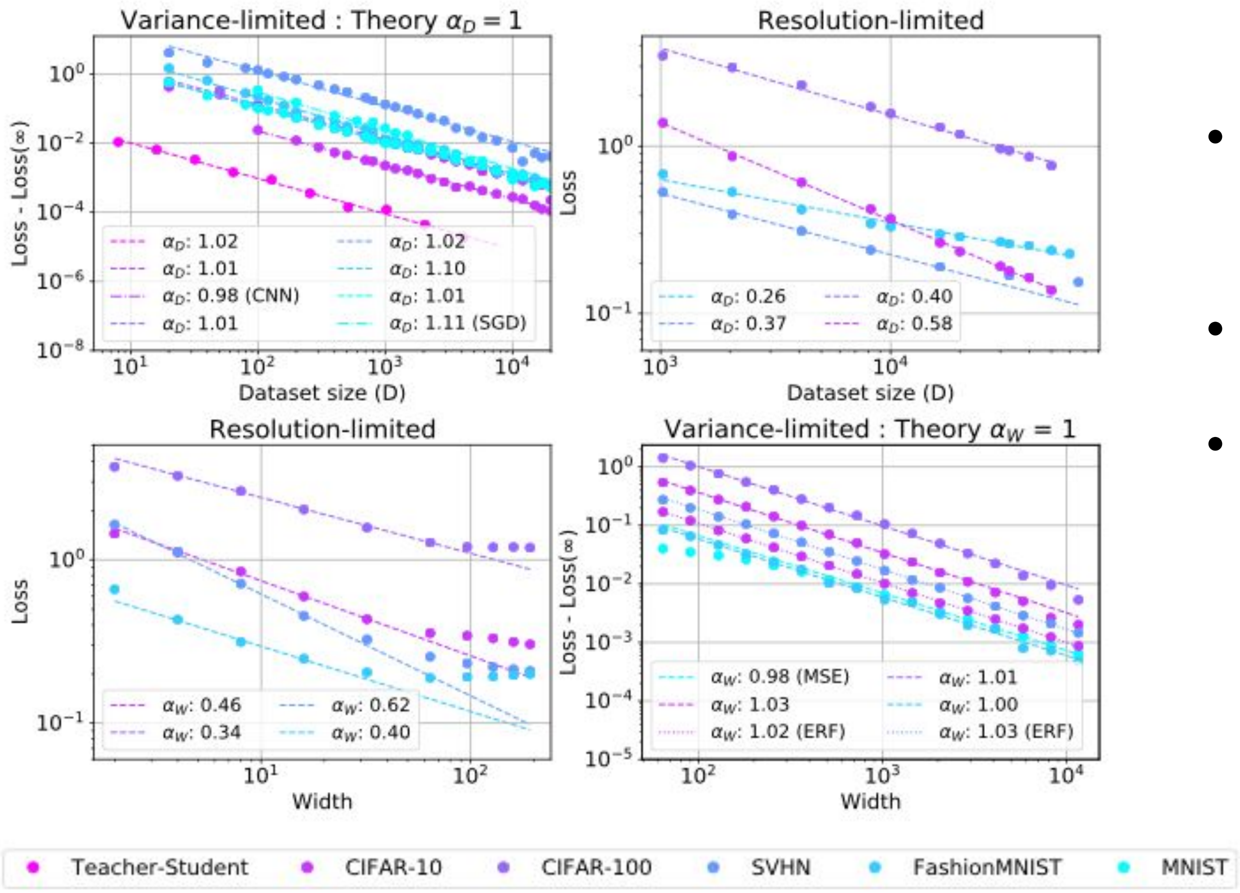
1. Fix D, let P >> D, study P scaling
2. Fix P, let D >> P, study D scaling

Resolution-limited regime: Take one of D or P to be effectively infinite (>>) so it is no longer a bottleneck and examine *scaling as a function of the other variable*.
Nontrivial (non-integer) exponents.

1. Fix P, P >> D, study D scaling
2. Fix D, D >> P, study P scaling

	Underparameterized D >> P	Overparameterized P >> D
Scaling with D	**Variance-limited**	**Resolution-limited**
Scaling with P	**Resolution-limited**	**Variance-limited**

Four regimes together (neural networks)



- Different datasets
  - CIFAR-10, CIFAR-100
  - SVHN
  - MNIST, Fashion-MNIST
  - Teacher-Student
- Different models
  - FC, CNN, WRN
  - Nonlinearities
- Different loss functions
  - MSE, Cross-entropy

Classification

Support for this comes from:

- Student-teacher random features setting
  - Explicit derivations for **all four regimes**
  - Can relate exponents to properties of kernel (**power law decay in eigenvalue spectrum**)
  - We also obtain a ***duality*** between model and dataset-size scaling (that is, exponents are the same).

We test these predictions empirically outside of this framework, in the pre-training + fine-tuning setting, and find agreement.

- General setting (model, data, …)
  - Variance-limited regime (**two regimes**): formal proof.
  - Resolution-limited regime (**two regimes**): posit an expansion & test empirically.

Student-Teacher Random Features Model

Concrete case study: student-teacher random features model

In these cases, we can derive variance-limited & resolution-limited exponents from exact expressions.

Setup we consider (MSE):

- Linear teacher model constructed from fixed features, $\{F_M\}$, potentially infinite.

$$\text{Teacher } F(x) = \sum_{M=1}^{S} \omega_M F_M(x) \qquad \text{Teacher weights } \omega_M \sim \mathcal{N}(0, 1/S)$$

- Student model constructed from features which are some projection of teacher features.

$$\text{Student } f(x) = \sum_{\mu=1}^{P} \theta_\mu f_\mu(x) \qquad f_\mu(x) = \textstyle\sum_M \mathcal{P}_{\mu M} F_M(x)$$

We analyze explicit expressions for the test loss in terms of the Gram & projection matrices, extracting the leading term.

Concrete case study: student-teacher random features model

Variance-limited exponent (=1):

Originates from fluctuations of finite covariance matrix about limiting covariance.

Resolution-limited exponent:

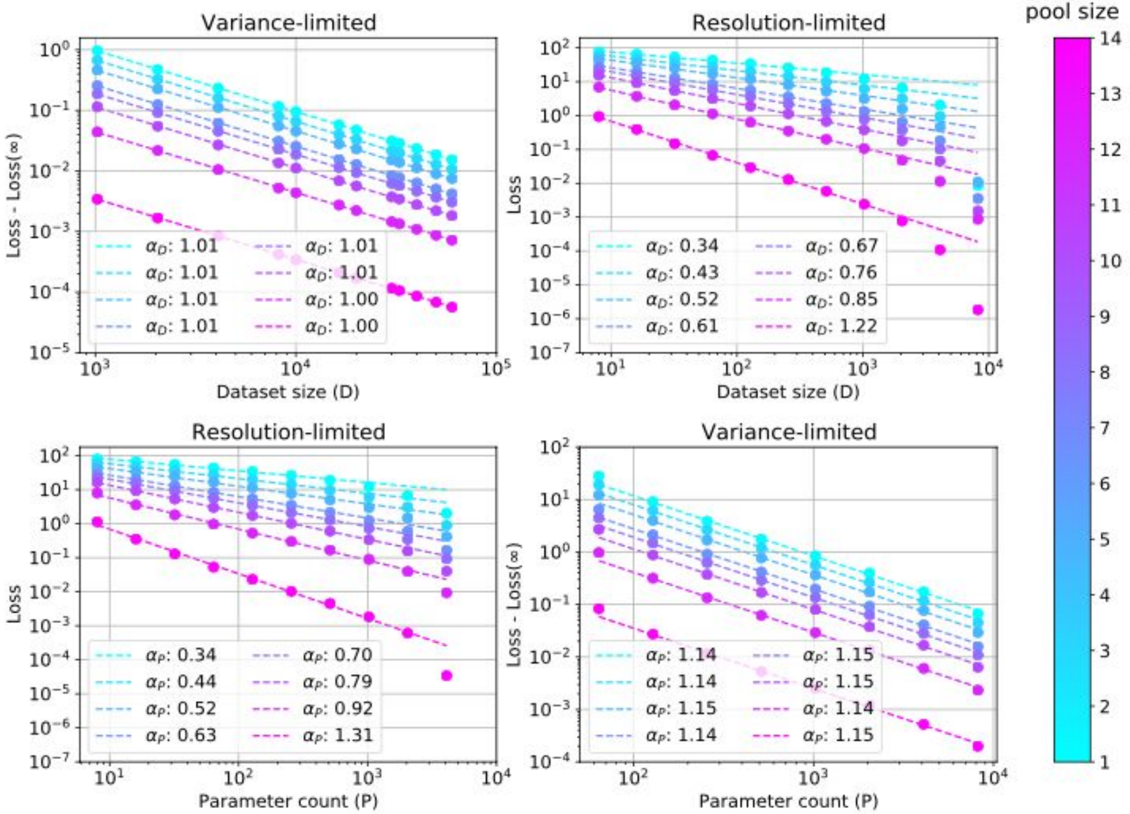
Analyze for eigenvalues satisfying power-law spectrum.

- Kernel generically has power-law spectrum with exponent 1+ α_K: $\lambda_i \sim \frac{1}{i^{1+\alpha_K}}$

$$\lambda_i \sim \frac{1}{i^{1+\alpha_K}} \Rightarrow \qquad L(D) \propto D^{-\alpha_K}, L(P) \propto P^{-\alpha_K}$$

- Then α_D, α_P = α_K. Note we also have a **“duality” ($\alpha_D = \alpha_P$)**: data and model scaling exponent is the same.

Concrete case study: student-teacher random features model

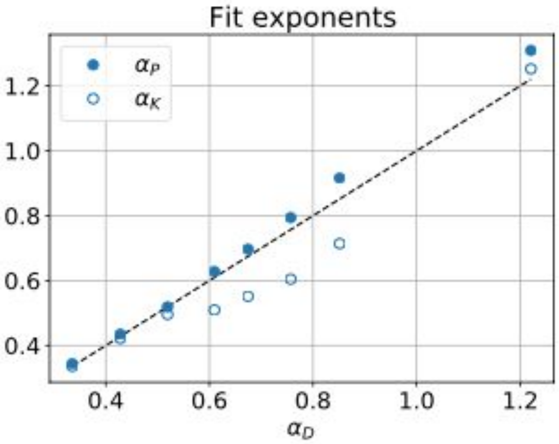


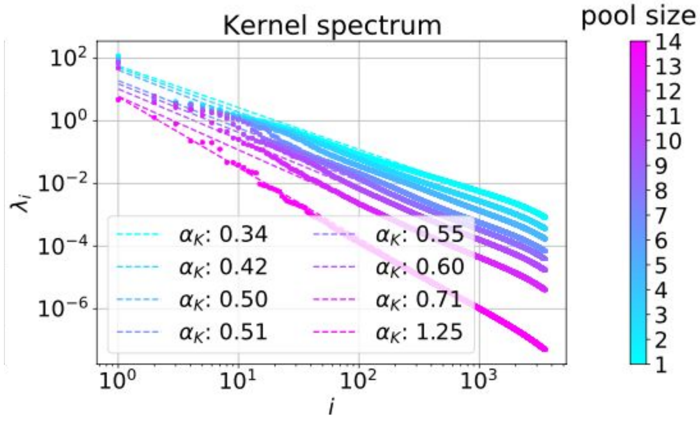
Linear student-teacher models with random features, trained with MSE loss to convergence.

Dataset is varied by downsampling MNIST by the specified pool size.

Concrete case study: student-teacher random features model

$$\lambda_i \sim \frac{1}{i^{1+\alpha_K}} \Rightarrow \qquad L(D) \propto D^{-\alpha_K}, L(P) \propto P^{-\alpha_K}$$





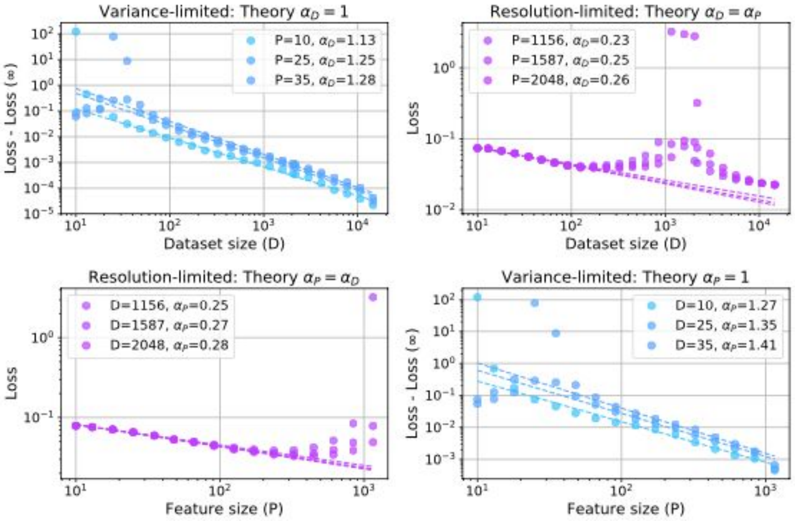
Kernel spectrum is for random fully-connected deep network (Relu) on pooled MNIST

Sufficiently smooth kernels have a bound on their eigenvalue spectrum.

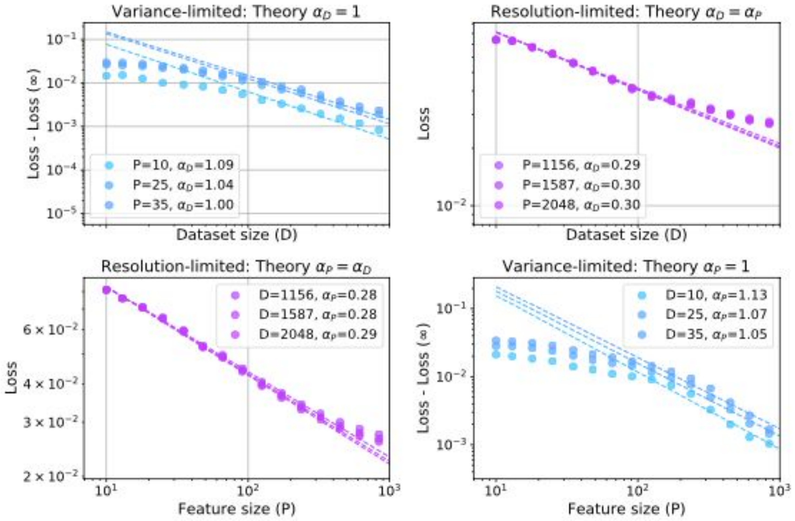
[Weyl; Kühn]. A C^t kernel on a *d*-dimensional space has eigenvalues: $$\lambda_n \lesssim \frac{1}{n^{1+t/d}}$$

Linear classifier from pre-trained embeddings

EfficientNet-B5 model pre-trained on ImageNet.
We use it to extract features for CIFAR-10 and then train linear classifier on these features.



Low regularization



Tuned regularization

Despite using ***actual CIFAR-10 targets***, we still observe **duality** in resolution-limited scaling ($\alpha_D = \alpha_P$).

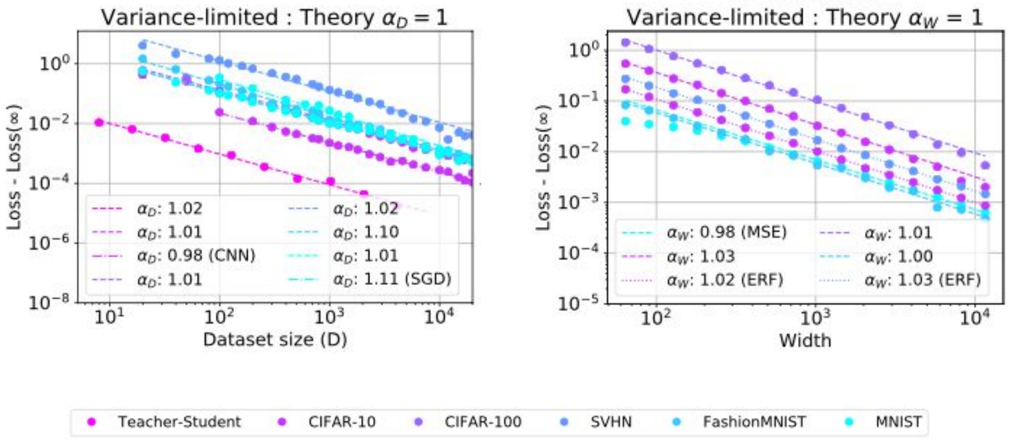
General Setting

Variance-limited scaling

See paper for details.

Large dataset behavior: concentration around population values (e.g. gradients of loss in gradient descent)

Large width behavior: based off of (known) leading finite-width corrections to infinite-width limit

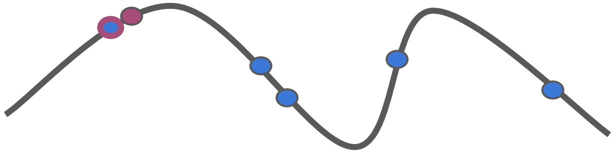


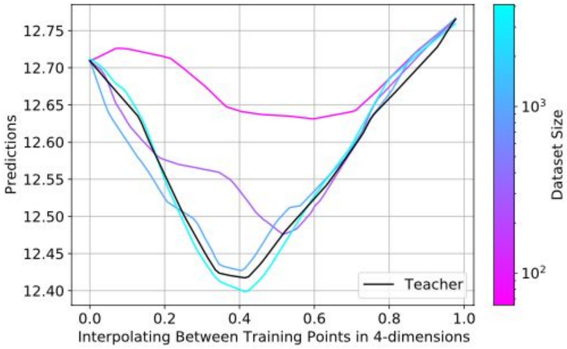
Resolution-limited scaling

Here, one variable taken -> infinity, study scaling with other variable.

At a rough level, the other variable is helping you better resolve a manifold.

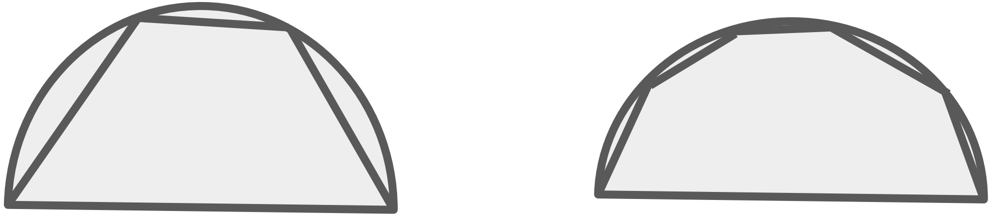
- Suppose P >> D (overparameterized): training points provide interpolation / anchor points and error at test point is controlled by distance to nearest training point.



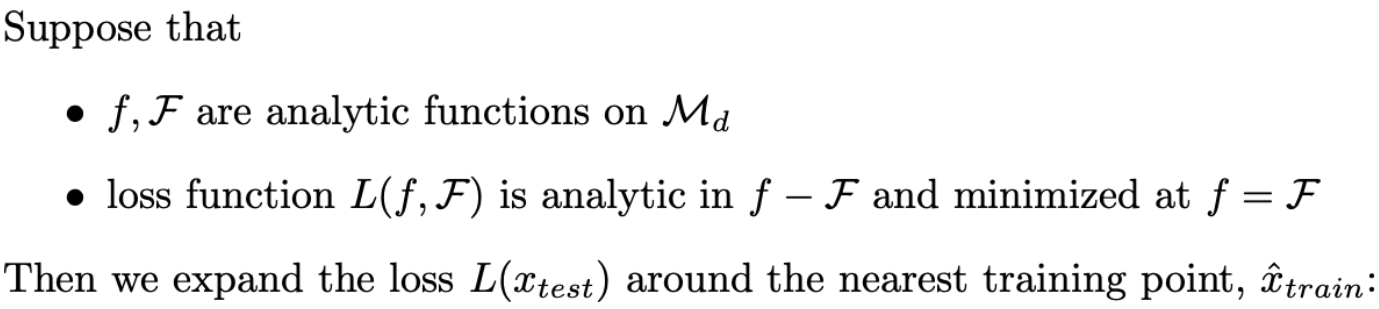


Interpolation in a student model as we add more data, in a 4d input space.

- Suppose D >> P (underparameterized). Why might "resolution-limited" picture be reasonable? Adding parameters gives you expressivity / degrees of freedom to fit the manifold.



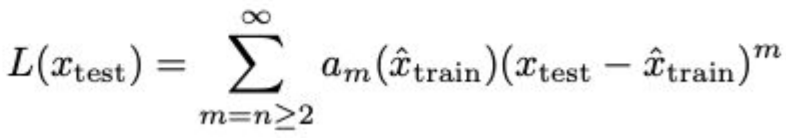
Resolution-limited scaling: series expansion



Suppose that

- $f, \mathcal{F}$ are analytic functions on $\mathcal{M}_d$
- loss function $L(f, \mathcal{F})$ is analytic in $f - \mathcal{F}$ and minimized at $f = \mathcal{F}$

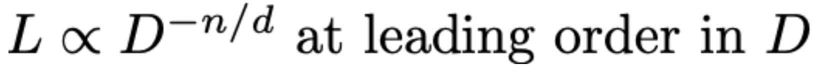
Then we expand the loss $L(x_{test})$ around the nearest training point, $\hat{x}_{train}$:



$$L(x_{\text{test}}) = \sum_{m=n\geq 2}^{\infty} a_m(\hat{x}_{\text{train}})(x_{\text{test}} - \hat{x}_{\text{train}})^m$$

Tied to scaling of nearest-neighbor distance on d-dimensional manifold.

Since the typical nearest neighbor distance scales as ~ $D^{-1/d}$ or as ~$P^{-1/d}$ we expect:

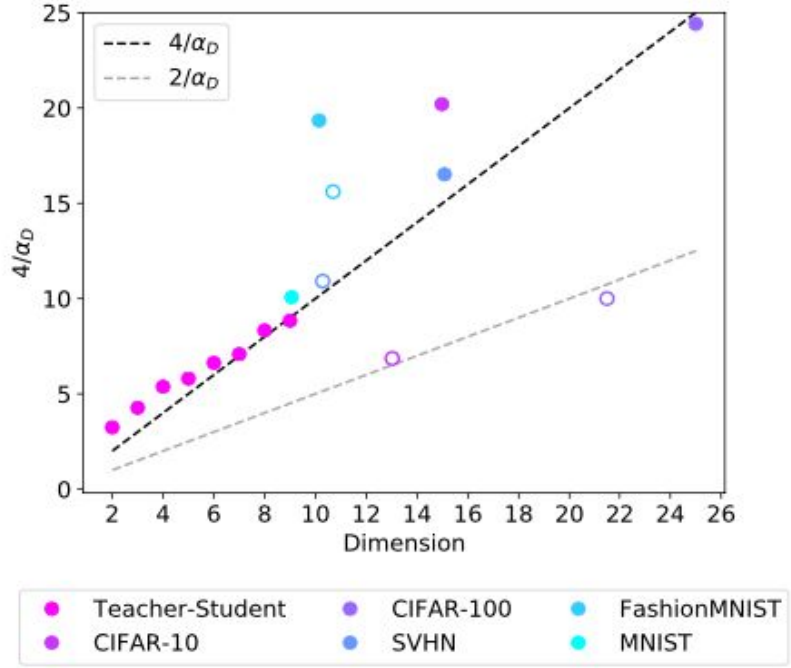


$L \propto D^{-n/d}$ at leading order in D

For a piecewise linear function, generically expect to start at n = 4.

Comparison between α_D and intrinsic dimensionality, d

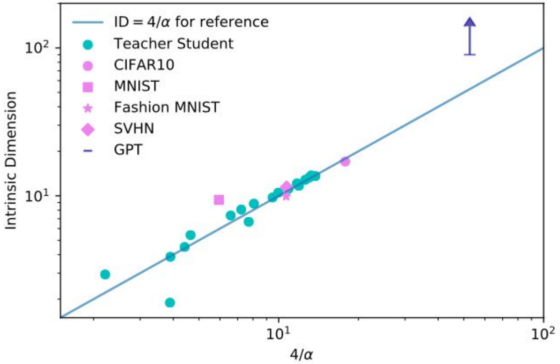
Dataset size scaling with dimension



Solid circles (CNN); hollow circles (Wide Resnet 28-10)

Left: estimating *d* by examining power-law scaling of nearest-neighbor distances from penultimate layer of trained network.

Model size scaling with dimension



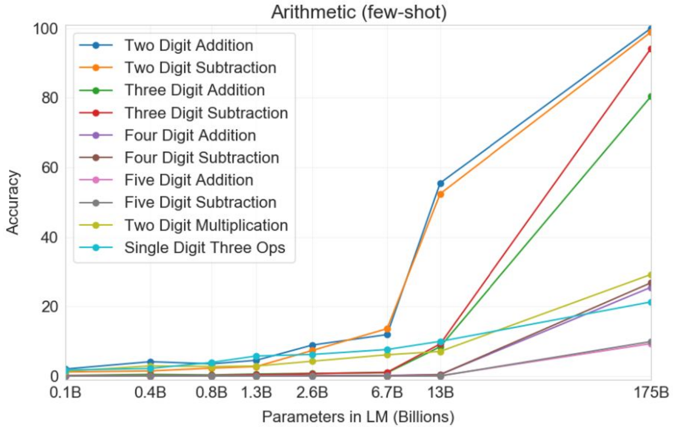
Sharma & Kaplan, arxiv 2004.10802

Conclusion

- Categorization of exponents based on mechanism
  - In some regimes, expect a common scaling exponent
  - In other regimes, get non-integer powers controlled by kernel eigenvalues, effective dimensionality

- Limitations
  - Lack control on intermediate regimes (D ~ P) or small values (small D, P)

- Largest language models trained on diverse data: have been observed to break scaling!



Brown, et al. arxiv 2005.14165 (2020).

See also Bordelon, et al. ICML 2020.

End