Empirical Studies in Deep Learning

Types of Empirical Studies in Deep Learning

Machine Learning Aspects, Studied for Deep Learning

- Generalization
- Robustness
- Optimization
- Active Learning

Deep Learning Aspects, Studied for Deep Learning

- Lottery Ticket Hypothesis
- Mode Connectivity
- DL Architectures

- Pretraining
- Double Descent
- Overparameterization
- Scaling Laws

Scaling Laws
in Machine Learning

Machine Learning

$$\text{Samples } \{z_1, \dots, z_n\} \longrightarrow \boxed{A} \longrightarrow \text{Model } f$$

How *good* is our learning algorithm A? Many choices…

- Input distribution: **Fixed dist** / average over a family / worst-case

  (empirical) *(theory)*

- Model evaluation: Loss function? **On dist** / off-dist? Downstream eval?

- Sample-size: n. **Fixed** / asymptotic?

Deep Learning

Samples $\{z_1, \dots, z_n\}$ $\longrightarrow \boxed{A} \longrightarrow$ Model f

Most practical papers: fixed distribution, fixed sample size n_0

Model	Top-1	Top-5
Sparse coding [2]	*47.1%*	*28.2%*
SIFT + FVs [24]	*45.7%*	*25.7%*
CNN	**37.5%**	**17.0%**

Table 1: Comparison of results on ILSVRC-2010 test set. In *italics* are best results achieved by others.

[Krizhevsky et al. 2012]

model	top-1 err.	top-5 err.
VGG-16 [41]	28.07	9.33
GoogLeNet [44]	-	9.15
PReLU-net [13]	24.27	7.38
plain-34	28.54	10.02
ResNet-34 A	25.03	7.76
ResNet-34 B	24.52	7.46
ResNet-34 C	24.19	7.40
ResNet-50	22.85	6.71
ResNet-101	21.75	6.05
ResNet-152	**21.43**	**5.71**

Table 3. Error rates (%, **10-crop** testing) on ImageNet validation.

[He et al. 2016]

Deep Learning

Samples $\{z_1, \dots, z_n\}$ $\longrightarrow$ $\boxed{A}$ $\longrightarrow$ Model f

Consider: performance as a function of n.

1. Individual algos: Care about more than just fixed n_0
   (Want continued improvement with n…)

2. Comparing algos / Model selection:
   - In future, with $n = 1e9$, which learning algo should we use?
   - Want *asymptotic behavior* at large n

Hope in DL:

1. (Large enough) Deep nets continue to improve with n

2. Algos which work best on ImageNet → best on larger problems
   (warning: until recently…)

Learning Curves

Fix distribution D, learning algo A. Define

$L(n)$:= *"expected test loss of A on* $\boldsymbol{n}$ *samples from D"*

$$= E_{S\sim D^n,A}\left[Loss_D\ A(S)\right]$$

L is known as the "learning curve" of A.

Long history in practice & theory…

> It is an important subject of research of neural networks and machine learning to study general characteristics of learning curves, which represent how fast the behavior of a learning machine is improved by learning from examples. It is also important to evaluate the performance of

[Amari, Murata 1993]

See also survey:
[Viering, Loog 2021]

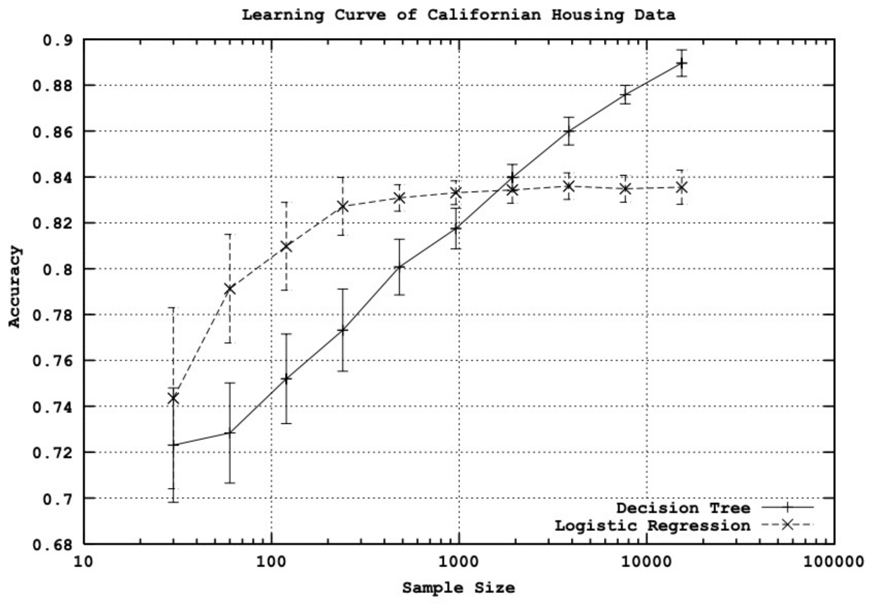
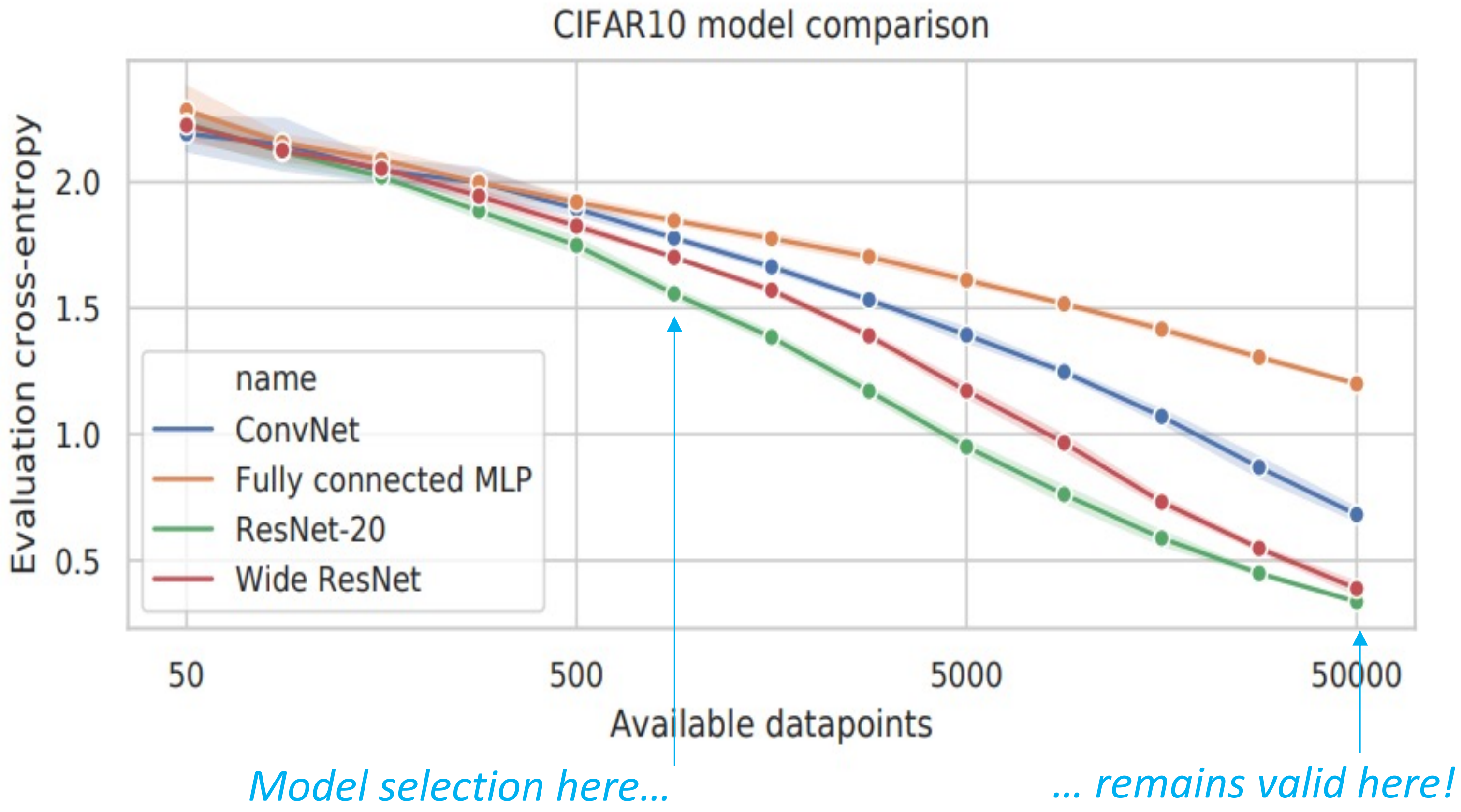


Figure 2: Log-scale learning curves

[Perlich Provost Simonoff 2001]

Typical Learning Curves in DL



Hope in DL:

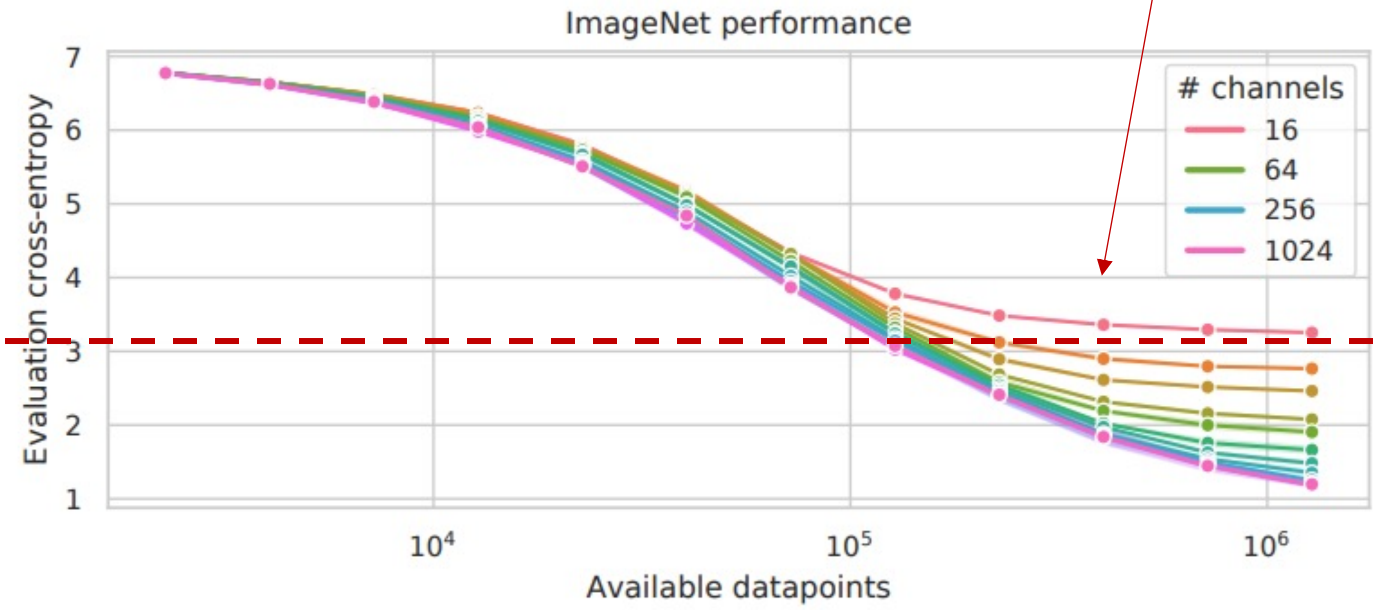
1. (Large enough) Deep nets continue to improve with n

2. Algos which work best on ImageNet → best on larger problems

[Bornschein Francesco Osindero 2020]

Why “Large Enough” NNs?

Small models “plateau”



$L(\infty)$

[Bornschein Francesco Osindero 2020]

Define “**reducible loss**” L^* :
Difference from “best possible” loss

$$L(n) = L^*(n) + L(\infty)$$

$L(\infty) > 0$ because:

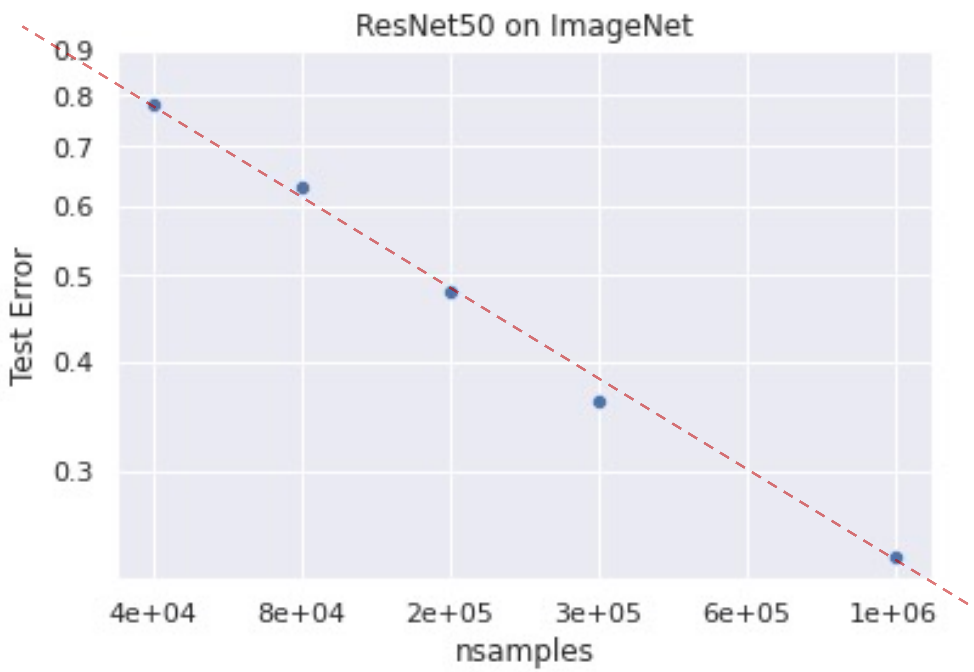
- Models too small
- Distribution inherently “noisy”

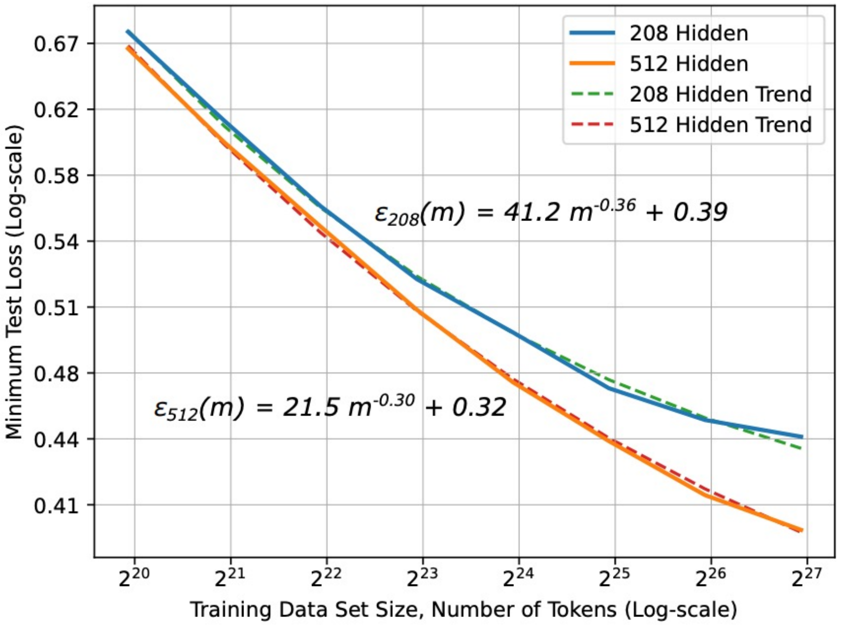
Power Law Scaling

<u>Claim</u>: For large-enough NNs, learning curve of reducible loss is a power law:

$$L^*(n) \sim An^{-\beta}$$

[Hestness et al. 2017]
[Rosenfeld, Rosenfeld, Belinkov, Shavit 2019]
[Kaplan, McCandlish et al. 2020]





Machine Translation: Fig 1, Hestness et al.

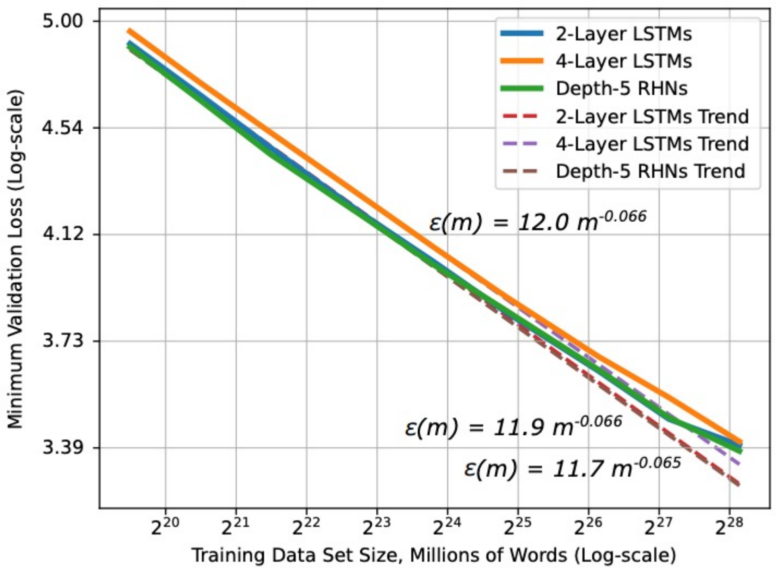
Scale Invariance

Power laws: $$L(n) = An^{-\beta}$$

$$\Updownarrow$$

Defining property: $L(kn) = C_k \, L(n)$

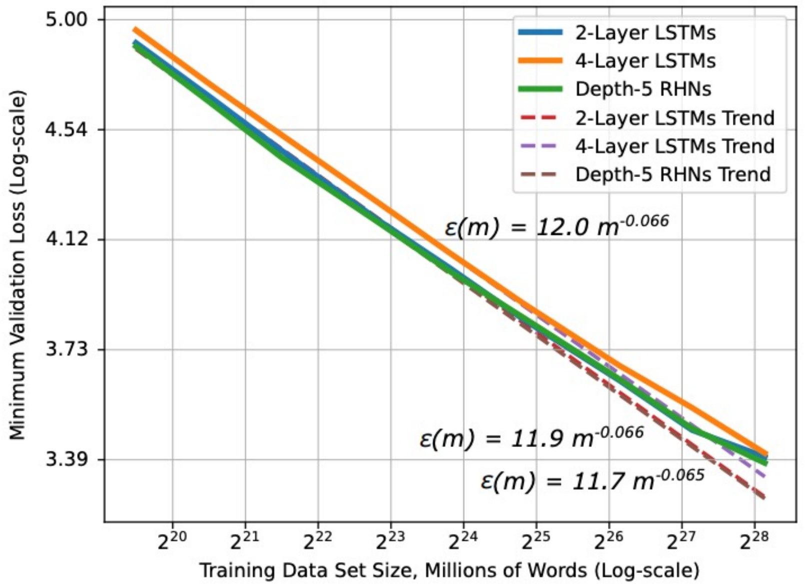
Eg: “Having **10x** times more data will reduce the loss by **50%**” (for all n)



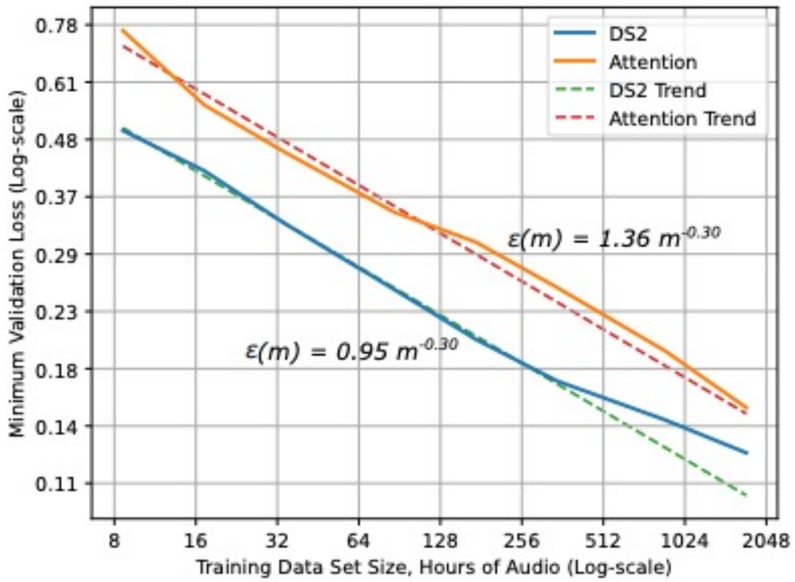
Power-law scaling verified in many settings:

- Domains: LM, MT, Text/Image classification, gen. modeling

- Architectures: ConvNets, Transformers, LSTMs, Kernels,…

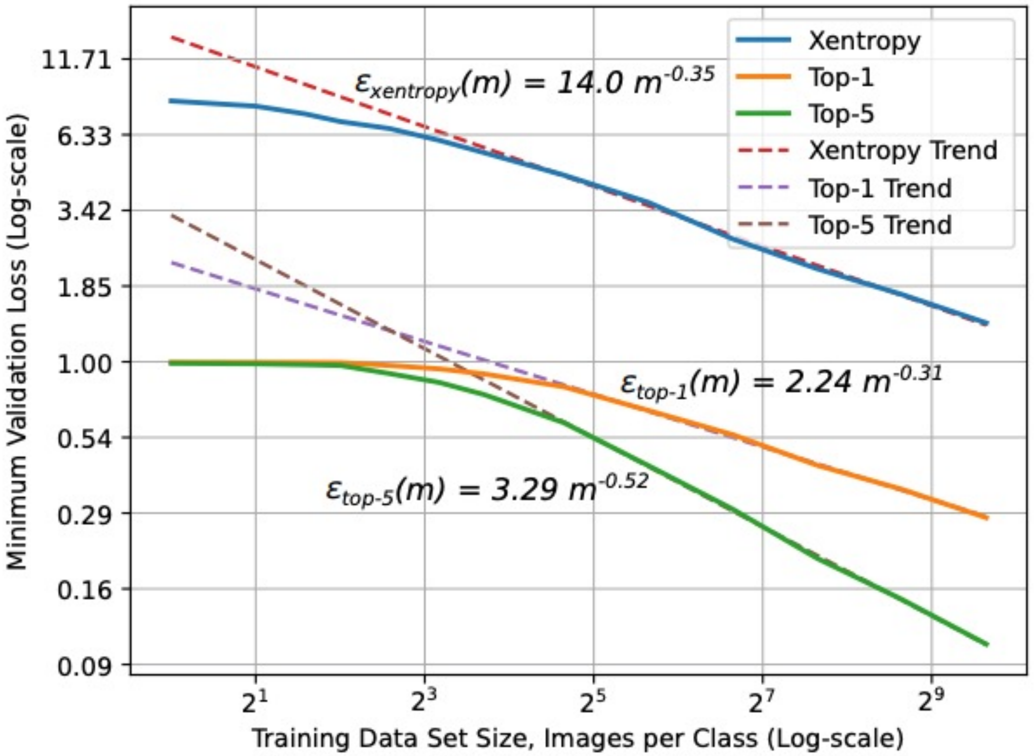


Language Modeling



Speech Recognition

Caveats: “Warm-up”



Transient region

Power-law region

Potential problem: We could be in the “transient” region without knowing it...

Consequences

1. Evaluate design choices in ML via effect on **scaling**:

constant (A) vs exponent (β)

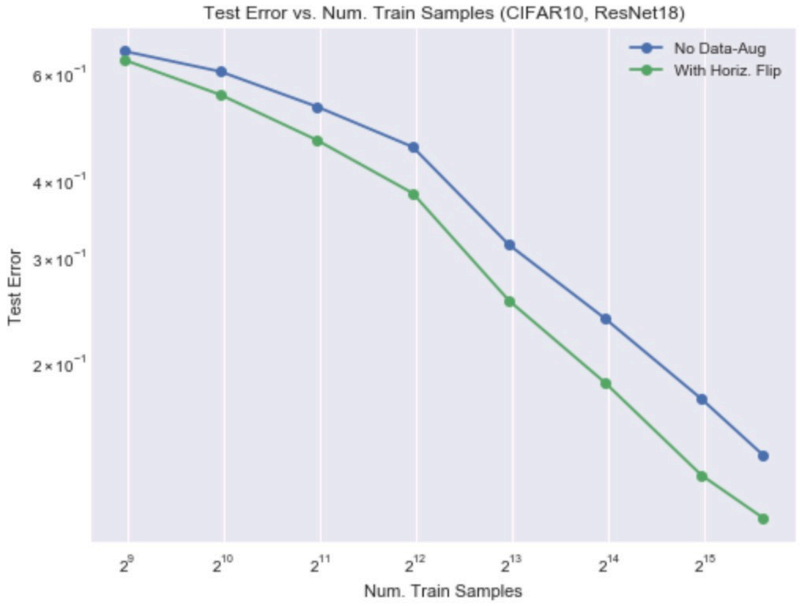
Ex: What is the effect of data-augmentation?
Likely only affects the constant (data: $n \mapsto Kn$). [Hoiem et al 2021]

Ex: What is the effect of architecture? (cf algo design…)

2. small scale experiments → large scale behavior

(good for science & practice… but caveats apply)

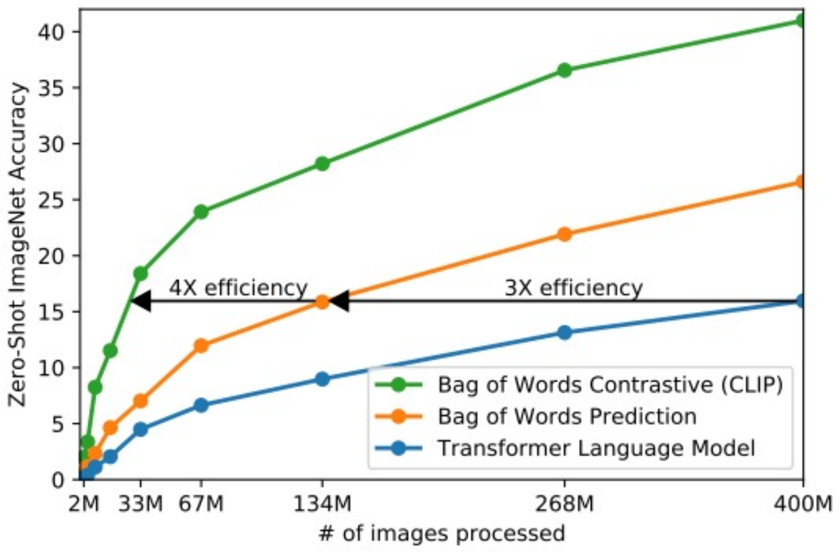
$$L^*(n) \sim An^{-\beta}$$

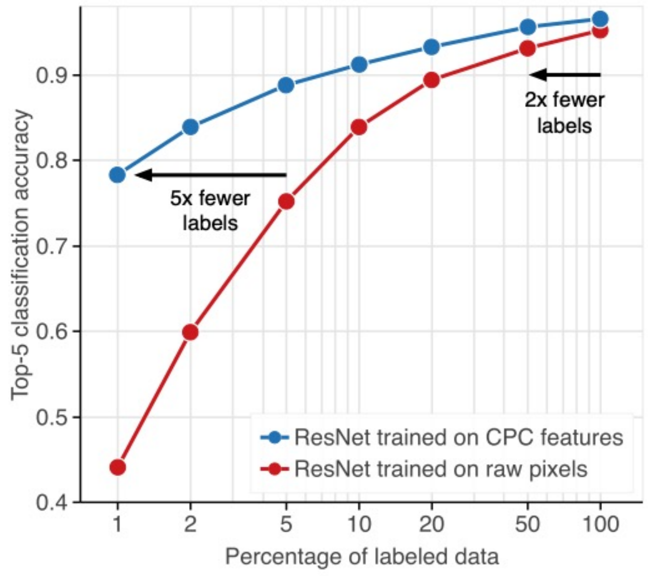


More common for papers to report data-scaling

(changes to the constants, not asymptotics…)

$$L(n) \sim An^{-\beta}$$





[Henaff et al. 2020]

s

Scaling: In Theory

Many upper-bounds in learning theory obey power-laws
Ex: ERM / Uniform convergence

$$f_n := \operatorname{argmin}_{f \in \mathcal{H}} \widehat{L_n}(f)$$
$$f^* := \operatorname{argmin}_{f \in \mathcal{H}} L(f)$$

$$L(f_n) \leq L(f^*) + O\left(\sqrt{\frac{VC(\mathcal{H})}{n}}\right)$$

$1/\sqrt{n}$ dependency: *statistical* reasons

Scaling: In Theory

Different mechanisms!

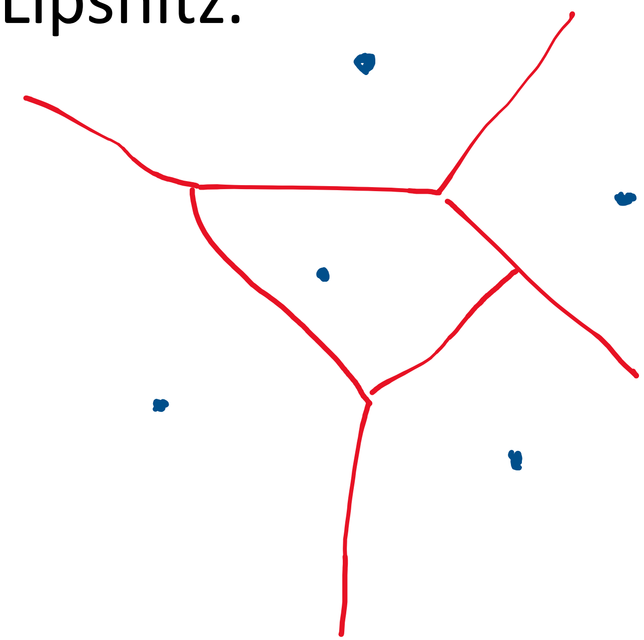
Algorithm/Setting	Rate	Notes
ERM	$O_{VC}(\frac{1}{\sqrt{n}})$	[SSS-SBD]
Parametric MLE	$O\left(\frac{d}{n}\right)$	[Liang]
SGD (online, convex)	$O\left(\frac{1}{t}\right)$	[Hazan] [Bottou]
GD (strongly convex)	$\exp(-\Omega(t))$	
1-NN classification	$O(n^{-\frac{1}{d}})$	[Chaudhuri-Dasgupta]
Kernel Smoothing (s-smooth)	$O(n^{-\frac{2s}{2s+d}})$	[Krishnamurthy]

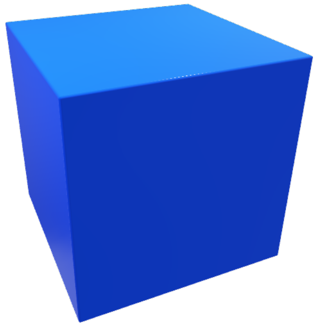
Scaling of 1-NN

Heuristic derivation based on
[Sharma, Kaplan 2020]

Regression: Want to estimate $f: [0,1]^d \to \mathbb{R}$,   1-Lipshitz.

n points, partition space into cells of sidelen s





Vol(cell) $\approx \frac{1}{n} \Longrightarrow s = n^{-1/d}$

Let $c(x)$: piecewise-const NN estimator

Loss (MSE):

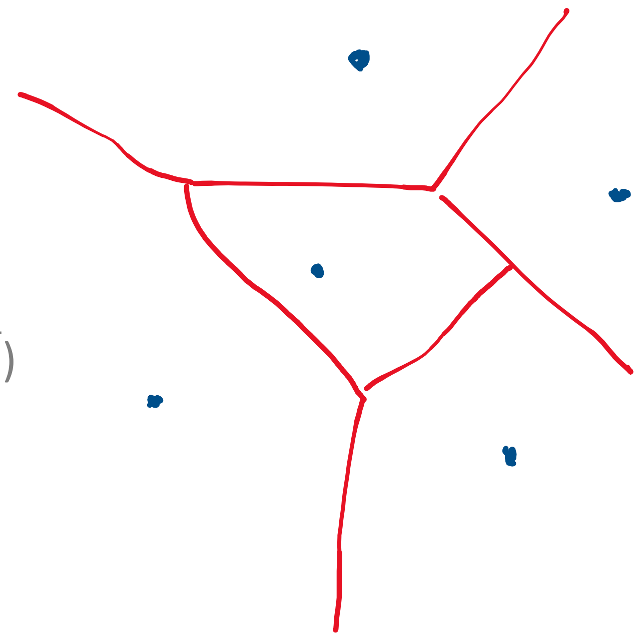
$$\mathrm{L} = \int_0^1 |f(x) - c(x)|^2 dV$$

$$\leq \int_0^1 \left|s\sqrt{d}\right|^2 dV$$

(Diameter of each cell $\sim s\sqrt{d}$)

$$\sim n^{-\frac{2}{d}}$$

$(s \sim n^{-1/d})$



Beyond Data-Scaling

Beyond Data-Scaling

Samples $\{z_1, \dots, z_n\}$ $\longrightarrow$ $\boxed{A_{S,T}}$ $\longrightarrow$ Model f

Specialize to neural-networks:

$L(N, S, T)$:= Test loss with **N** samples, model size **S**, train time **T**

N: info-theoretic constraint

S, T: computational constraint

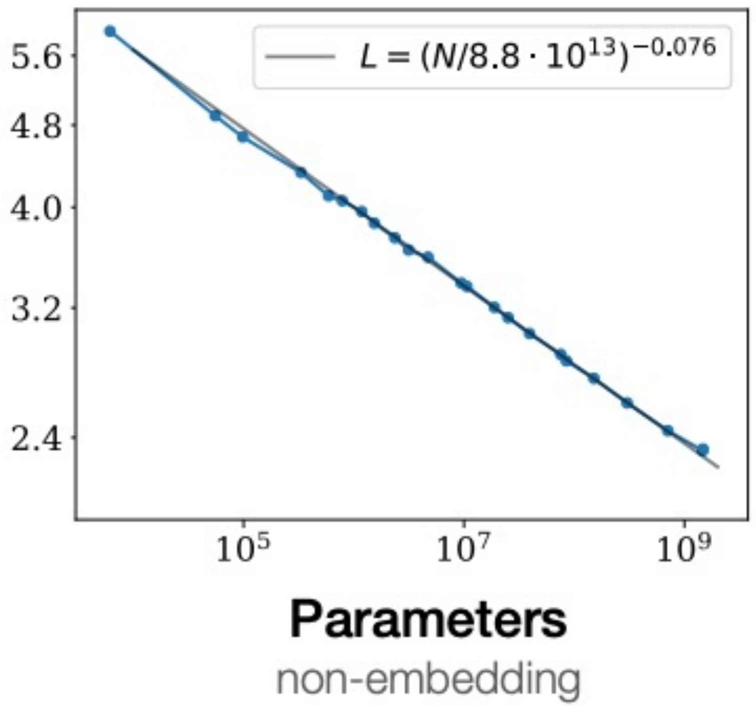
Well-behaved Regimes: PART I

$L(N, S, T)$

$L(\infty, \infty, \infty) \to 0$ *(or Bayes risk)*

$L(N, \infty, \infty)$: power-law data-scaling
$L(\infty, S, \infty)$: power-law model-scaling
$L(\infty, \infty, T)$: power-law online learning

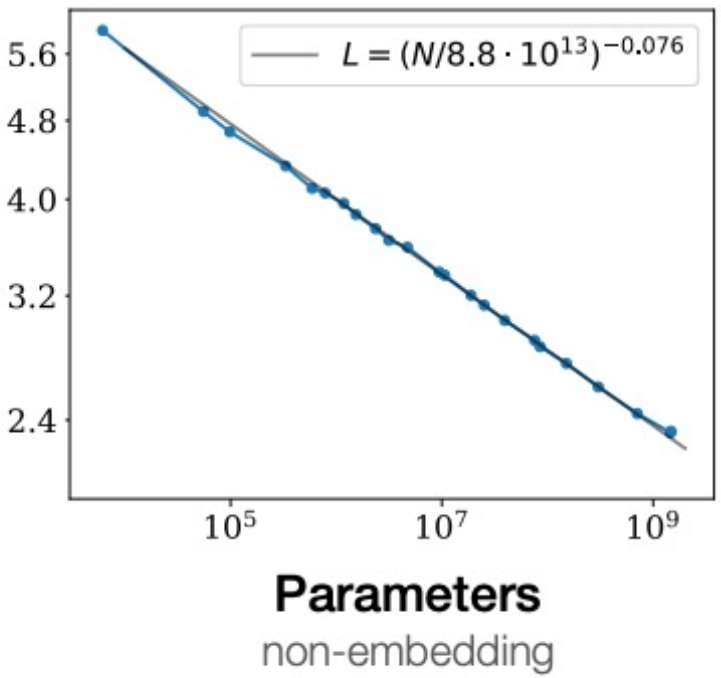
Bottlenecked by a single quantity.
“Resolution limited” [Bahri Dyer Kaplan Lee Sharma 2021]

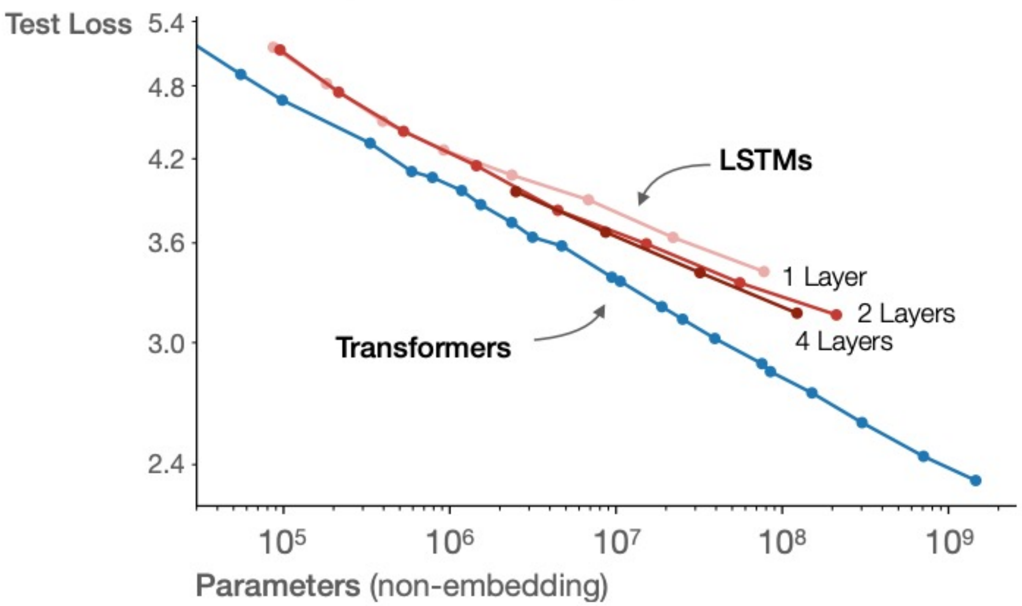


Model Scaling: $L(N = \infty, S, T = \infty)$

What is model "size"? Need a parameterization (width-scaling, etc)

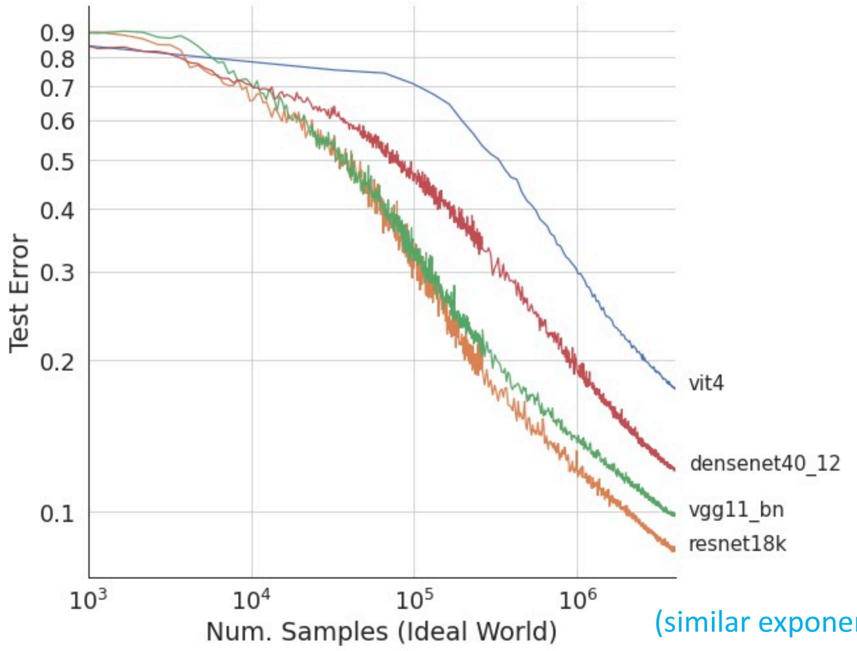
Generally, anything s.t. $L(\infty, \infty, \infty) \to 0$ will work





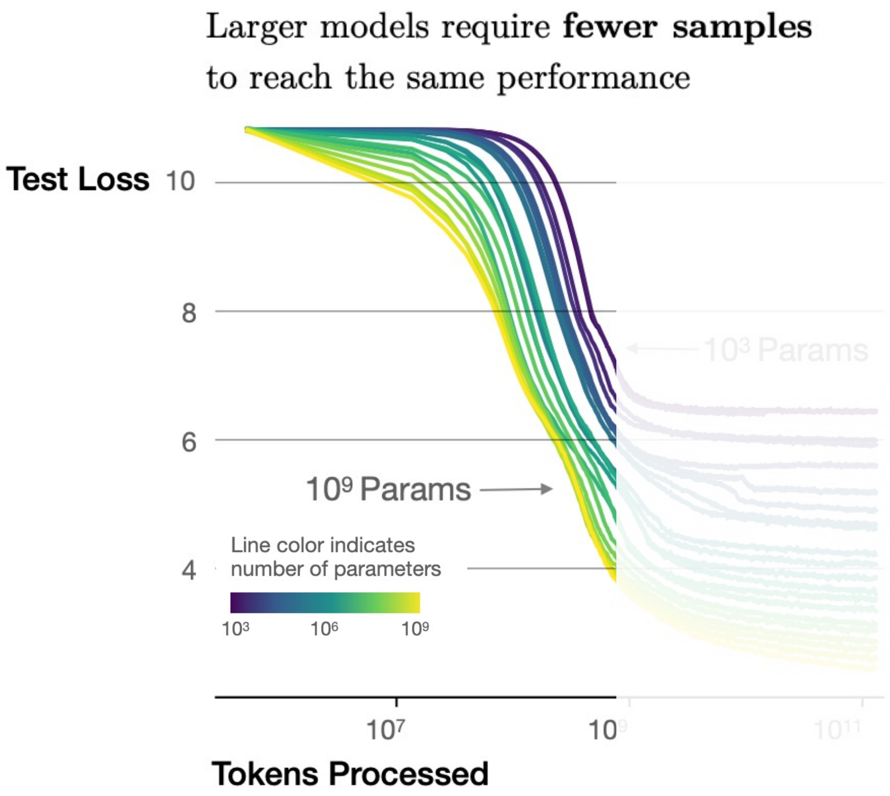
Online Learning Scaling: $L(N = \infty, S = \infty, T)$

"Effectively infinite data" = online learning



(similar exponents?)

CIFAR-5m



Language Modeling [Kaplan et al]

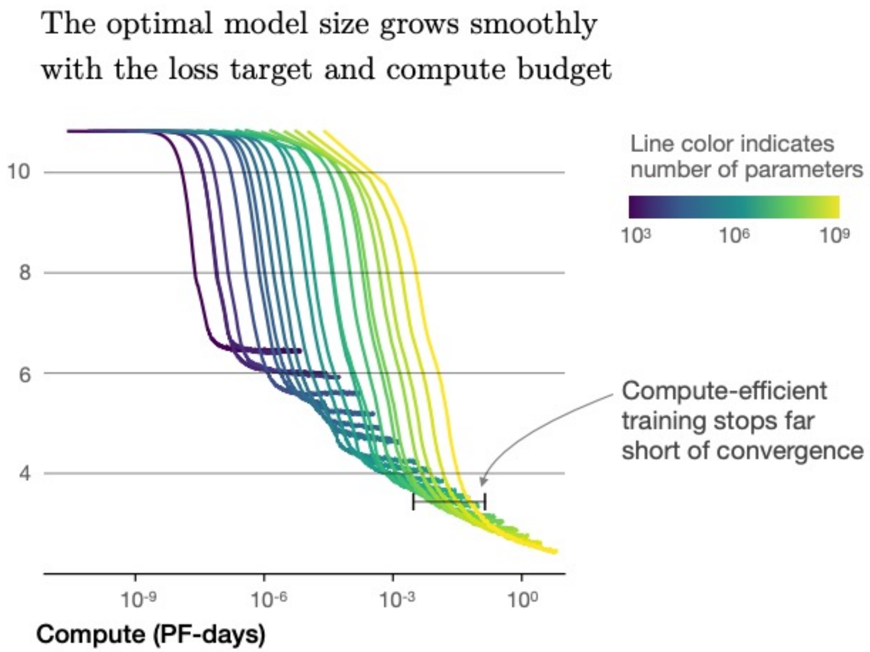
Compute-Scaling

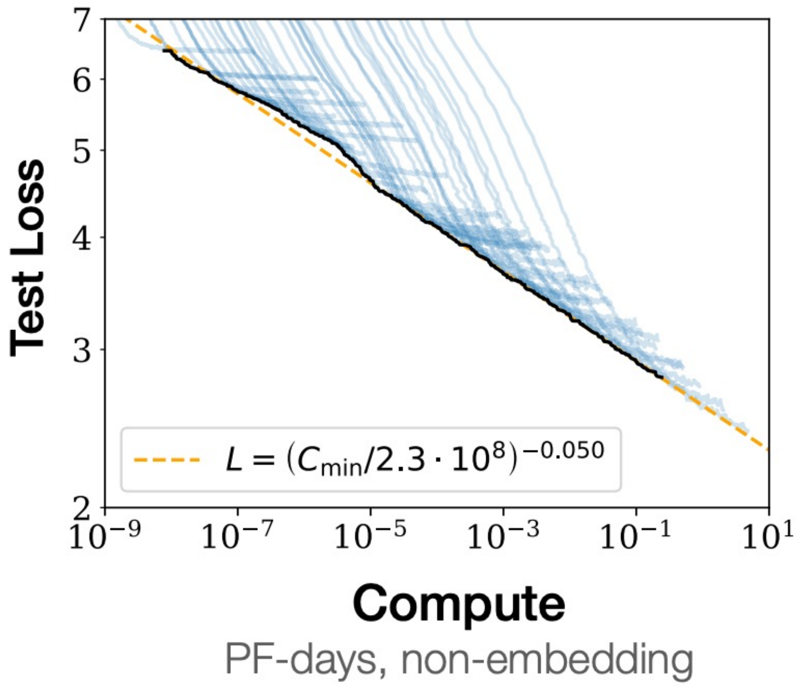
Practical measure: compute C, the cost of training (FLOPS)

Assume C ≈ S*T (depends on the parameterization of S)

Want: Optimal S, T within compute budget C (and infty data)

$$L_c(C) \coloneqq \min_{ST \leq C} L(\infty, S, T)$$





$$L_c(C) := \min_{ST \leq C} L(\infty, S, T) \approx L(\infty, C^\gamma, C^\delta)$$

Optimal S*, T* also follow power laws

Well-behaved Regimes: PART II

"Variance limited" regimes of $L(N, S, T)$ [Bahri Dyer Kaplan Lee Sharma 2021]

1. $L(N, S_0, \infty)$: power-law data-scaling for *reducible* loss ($N \gg S_0$)

irreducible loss = $L(\infty, S_0, \infty)$

2. $L(N_0, S, \infty)$: power-law model-scaling for *reducible* loss ($N_0 \ll S$)

irreducible loss = $L\left(N_{0,} \infty, \infty\right)$

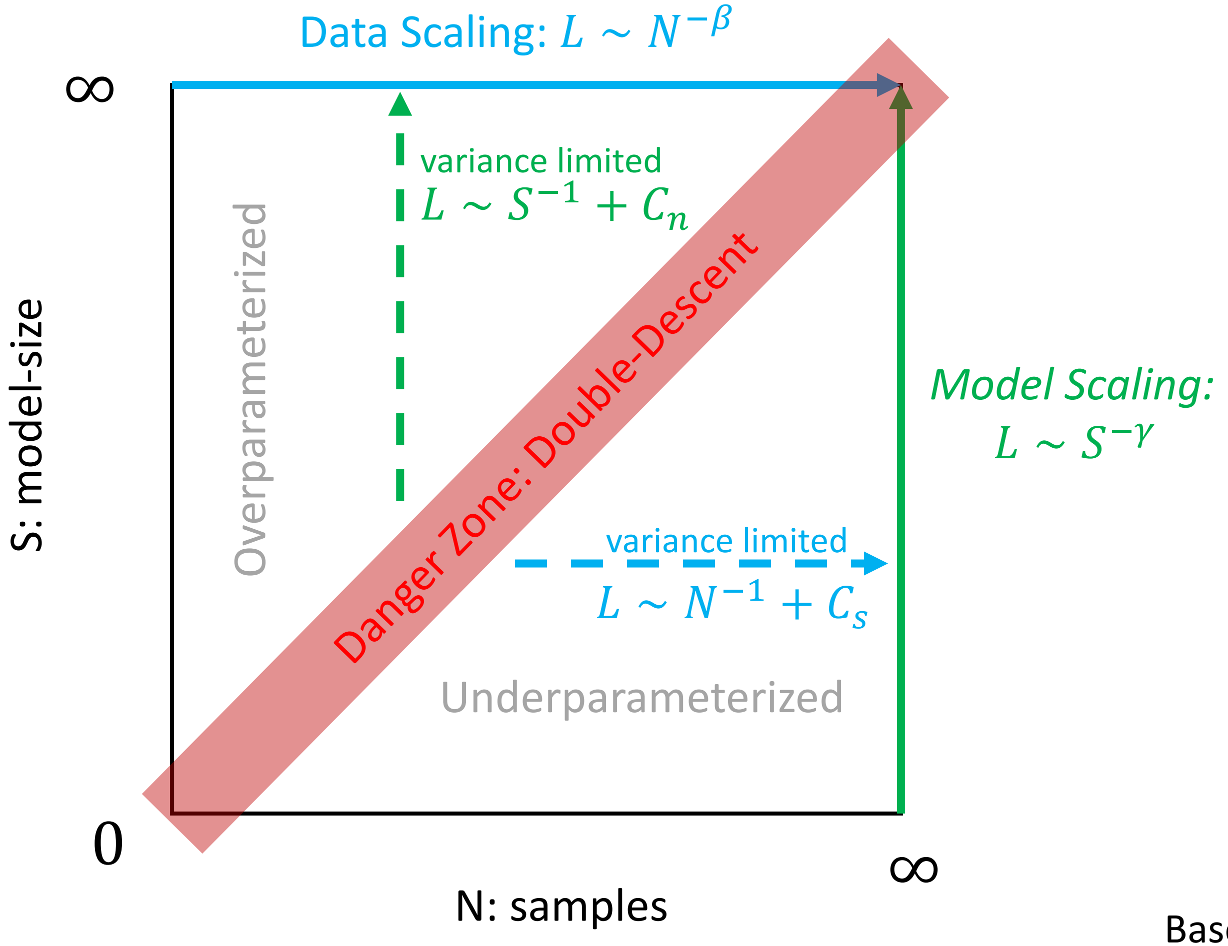
Power laws for *very different* reasons vs. earlier:

(1) is underparameterized, scales as $1/N$ for "classical" reasons (variance)

(2) also scales as $1/S$ (for similar reasons in certain cases)

Well-behaved Regimes: PART II

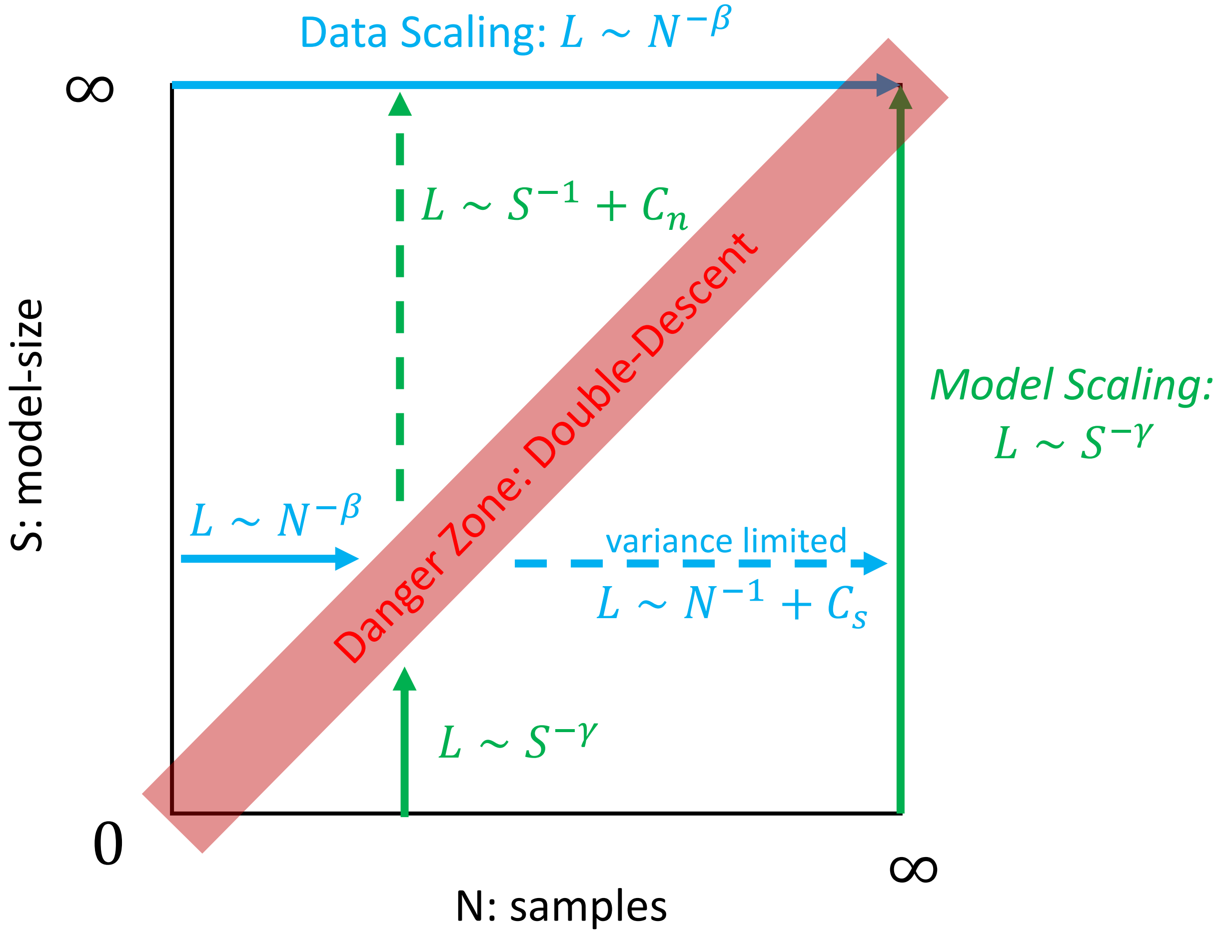
$L(N, S, T = \infty)$:

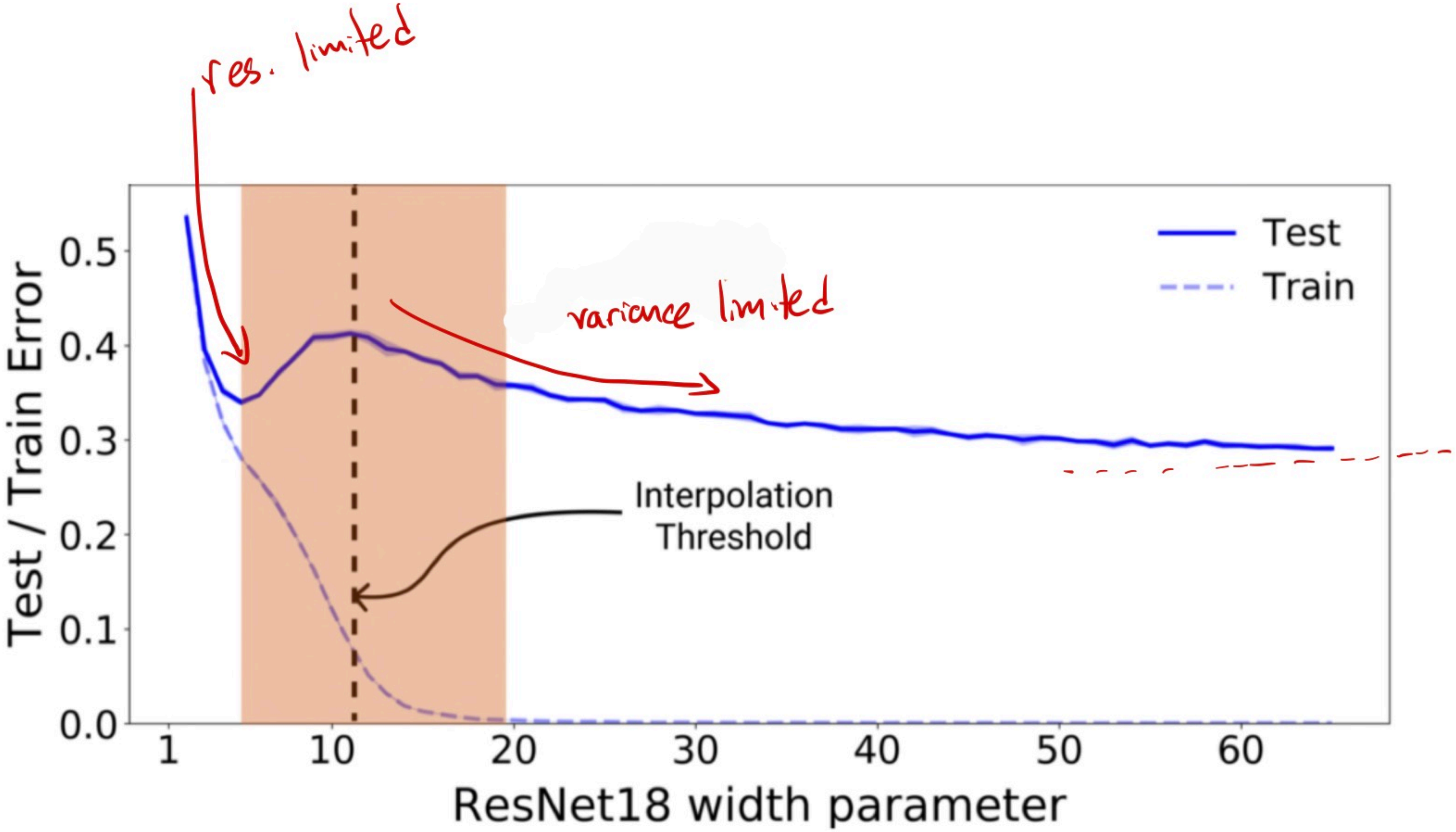


Based on
[Bahri Dyer Kaplan Lee Sharma 2021]

Well-behaved Regimes: PART II

$L(N, S, T = \infty)$:





For fixed N:

- First descent is "resolution limited" scaling
- Second descent is "variance limited" scaling

(edited 5/23/2021)

What Affects Scaling Exponent?

* Jury still out…

Architecture

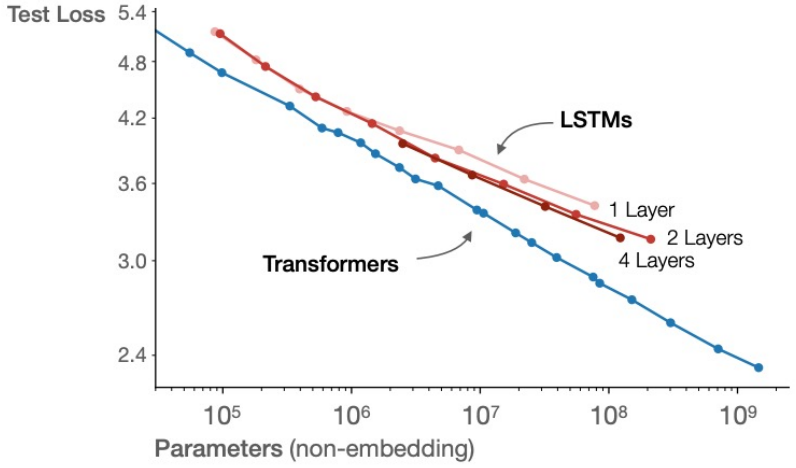
- Arch matters:
Exist architectures with bad scaling exponents (MLPs)

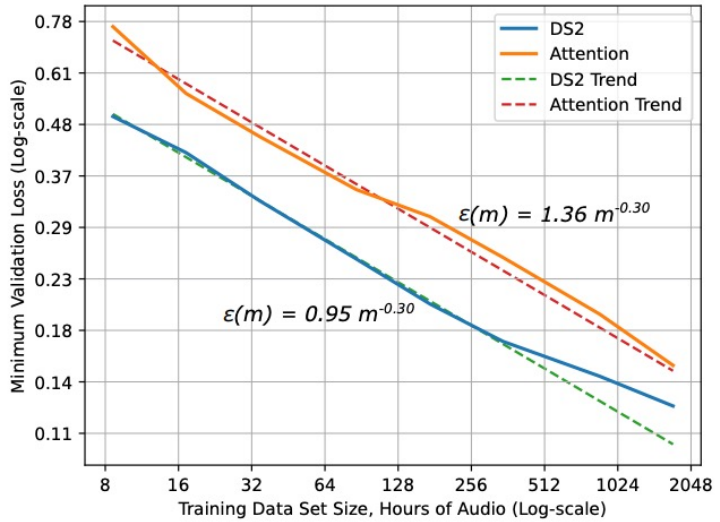
- **Arch doesn't matter**:

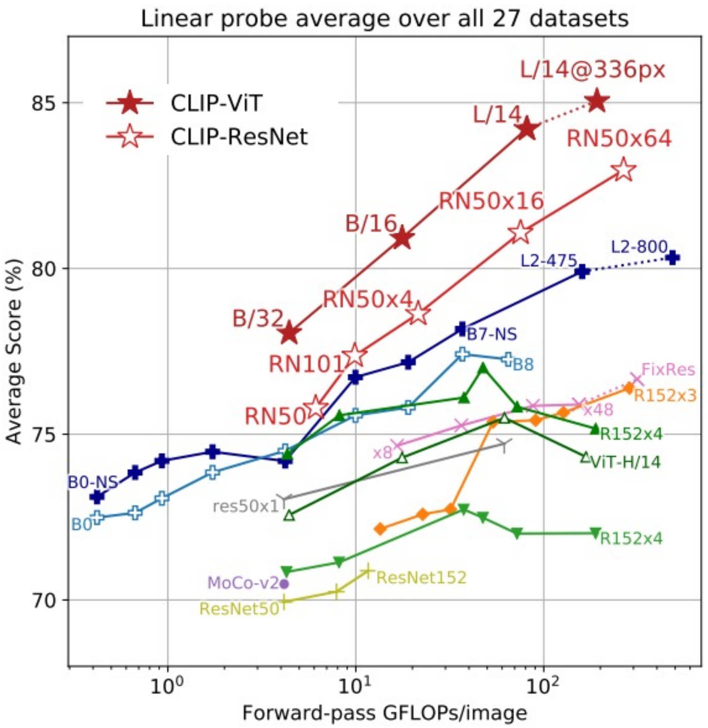
All "good" architectures have similar data/model/compute-scaling exponents

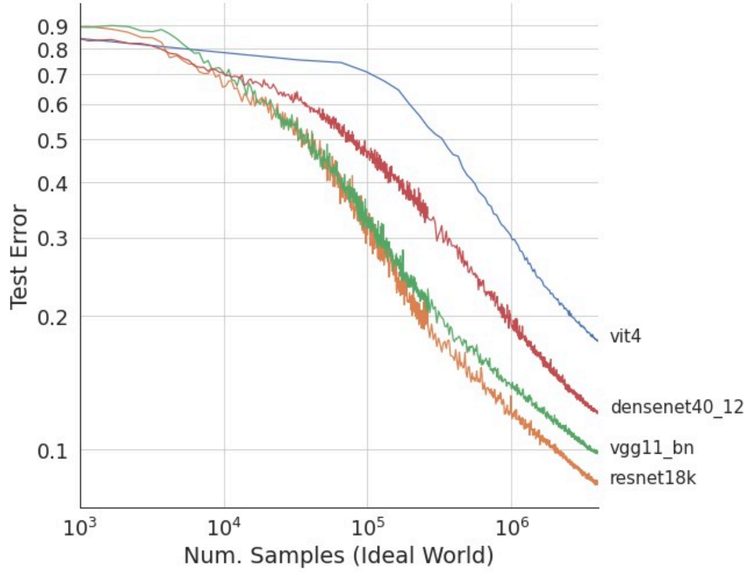
Even very different archs!

*don't know how to state this formally. could be wrong…





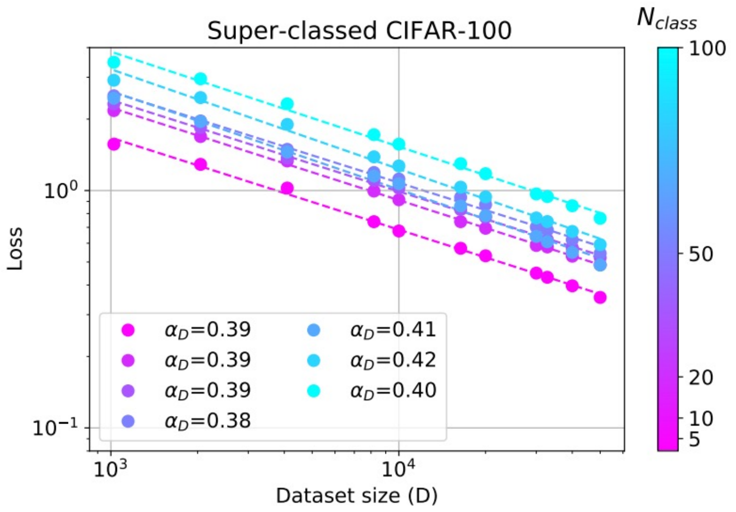




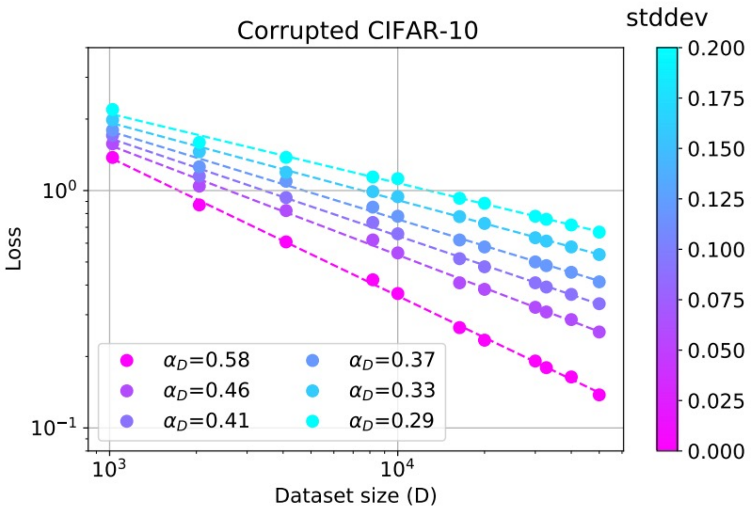
Data Distribution

Task matters, but not in obvious ways.   Eg: “Easier tasks have larger exponents?”

Two different ways to make task easier:



Superclassing: Same exponent



Adding feature noise: Changes exponent

Caveats

Loss vs Capabilities

Is all of DL predictable?

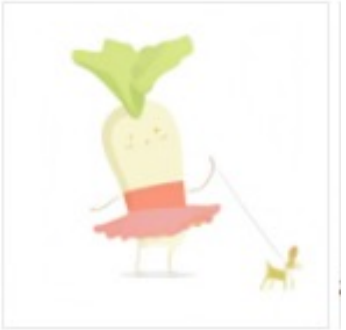
No. We're still surprised what happens "at scale"

Eg: ViT, DALL-E, GPT-3 few shot

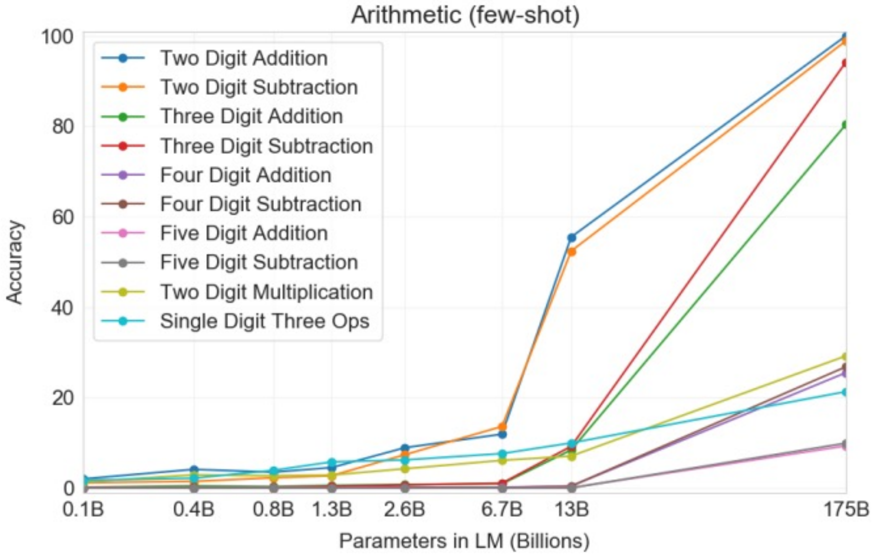
Why?

0. "Transient" effects
1. Don't know what to measure – some capabilities only appear "at scale"
2. Some measurements discontinuous

AI-GENERATED IMAGES



[Ramesh et al 2021]



Thanks!