Generalization Theory for Deep Learning

Rademacher Complexity

Intuition: how well can a classifier class **fit random noise?**

(Empirical) **Rademacher complexity:** For a training set $S = \{x_1, x_2, \ldots, x_n\}$, and a class $\mathcal{F}$, denote:

$$\hat{R}_n(S) = \mathbb{E}_\sigma \sup_{f \in \mathcal{F}} \sum_{i=1}^{n} \sigma_i f(x_i) \, .$$

where $\sigma_i \sim \text{Unif}\{+1, -1\}$ (Rademacher R.V.).

(Population) **Rademacher complexity:**

$$R_n = \mathbb{E}_S \left[\hat{R}_n(s) \right].$$

Rademacher Complexity Generalization Bound

Theorem: with probability $1 - \delta$ over the choice of a training set, for a bounded loss ℓ, we have

$$\sup_{f \in \mathcal{F}} \left| \frac{1}{n} \sum_{i=1}^{n} \ell(f(x_i), y_i) - \mathbb{E}_{(x,y)\sim D} \left[\ell(f(x), y) \right] \right| = O\left(\frac{\hat{R}_n}{n} + \frac{\log 1/\delta}{n} \right)$$

and

$$\sup_{f \in \mathcal{F}} \left| \frac{1}{n} \sum_{i=1}^{n} \ell(f(x_i), y_i) - \mathbb{E}_{(x,y)\sim D} \left[\ell(f(x), y) \right] \right| = O\left(\frac{R_n}{n} + \frac{\log 1/\delta}{n} \right)$$

Norm-based Rademacher complexity bound

Theorem: If the activation function is σ is ρ-Lipschitz. Let

$$\mathcal{F} = \{x \mapsto W_{H+1}\sigma(W_h\sigma(\cdots\sigma(W_1x)\cdots), \|W_h^T\|_{1,\infty} \leq B \forall h \in [H]\}$$

then $R_n(\mathcal{S}) \leq \|X^\top\|_{2,\infty}(2\rho B)^{H+1}\sqrt{2\ln d}$ where $X = [x_1, \ldots, x_n] \in \mathbb{R}^{d\times n}$ is the input data matrix.

Massart Lemma

Lemma: Let V be a set of vectors $\subset \mathbb{R}^d$, we have

$$R_V := \mathbb{E}_{\epsilon \sim \text{Unif}\{1,-1\}^d} \max_{v \in V} \langle \epsilon, v \rangle \leq \sup_{v \in V} \|v\|_2 \sqrt{2 \ln d}.$$

Properties of Rademacher Complexity

Let V be a set of vectors $\subset \mathbb{R}^d$:

1. $R_{\text{Conv}(V)} = R_V$,
2. $R_V = R_{-V}$.

3. Let $V_1, \ldots, V_m$ be m set of vectors such that for anyway $\epsilon \in \{-1, +1\}^d$, $\sup_{v \in V_i} \langle v, \epsilon \rangle \geq 0$ (e.g., $0 \in V_i$), then we have

$$R_{\cup_{i=1}^m V_i} \leq \sum_{i=1}^{m} R_{V_i}.$$

Proof of $(1,\infty)$ norm-based bound

Proof of $(1,\infty)$ norm-based bound

Proof of $(1,\infty)$ norm-based bound

Comments on generalization bounds

- When plugged in real values, the bounds are rarely non-trivial (i.e., smaller than 1)
- "*Fantastic Generalization Measures and Where to Find them*" by Jiang et al. '19 : large-scale investigation of the correlation of extant generalization measures with true generalization.

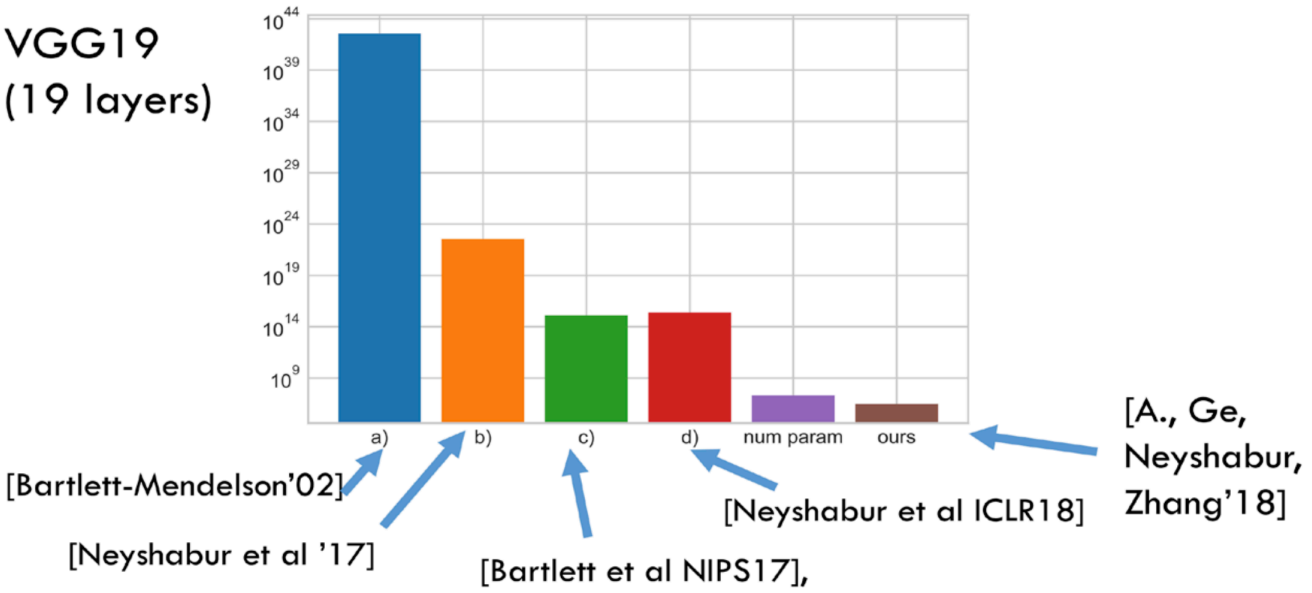


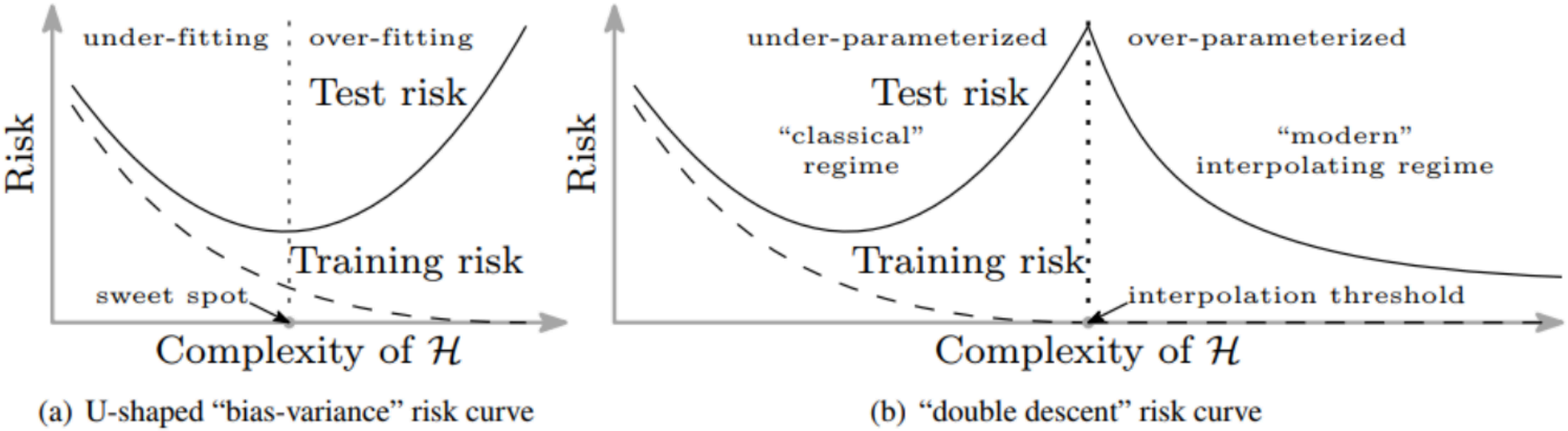
Image credits to Andrej Risteski

Comments on generalization bounds

- Uniform convergence may be unable to explain generalization of deep learning [Nagarajan and Kolter, '19]
  - Uniform convergence: a bound for all $f \in \mathcal{F}$
  - Exists example that 1) can generalize, 2) uniform convergence fails.

- Rates:
  - Most bounds: $1/\sqrt{n}$.
  - Local Rademacher complexity: $1/n$.

Double descent



(a) U-shaped "bias-variance" risk curve

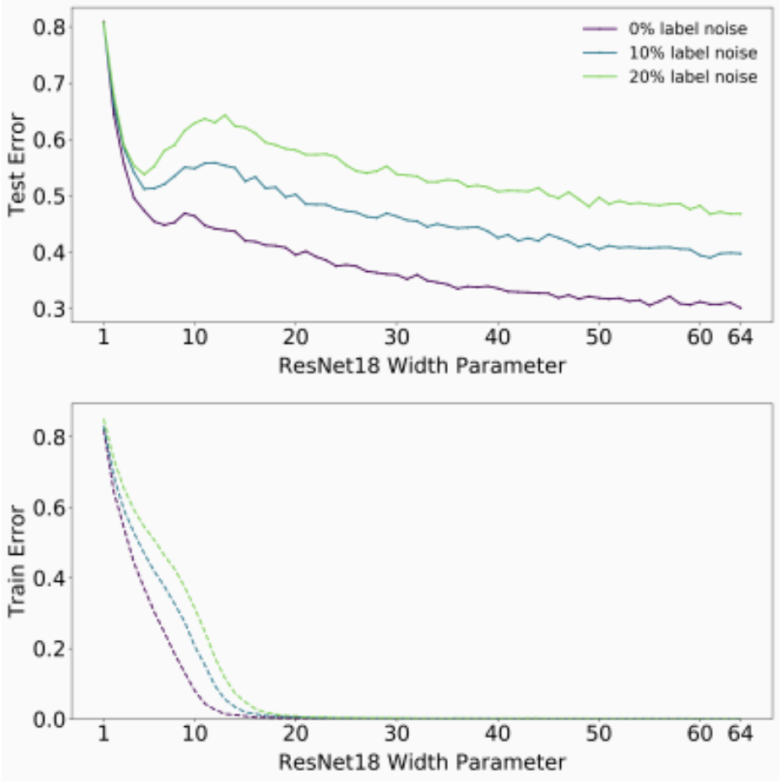
(b) "double descent" risk curve

Belkin, Hsu, Ma, Mandal '18

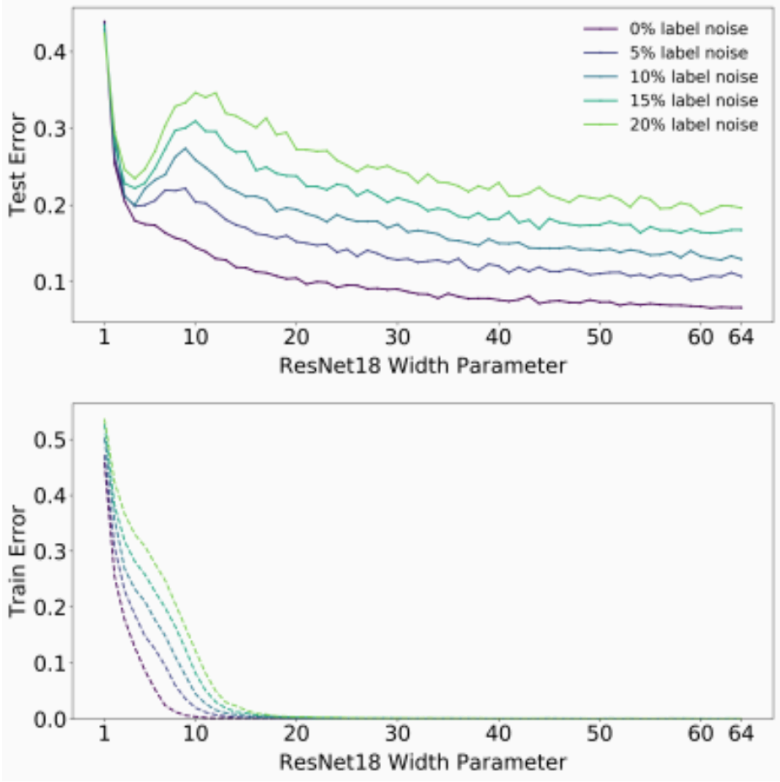
- There are cases where the model gets bigger, yet the (test!) loss goes down, sometimes even lower than in the classical "under-parameterized" regime.
- Complexity: number of parameters.

Double descent

Widespread phenomenon, across architectures (Nakkiran et al. ’19):



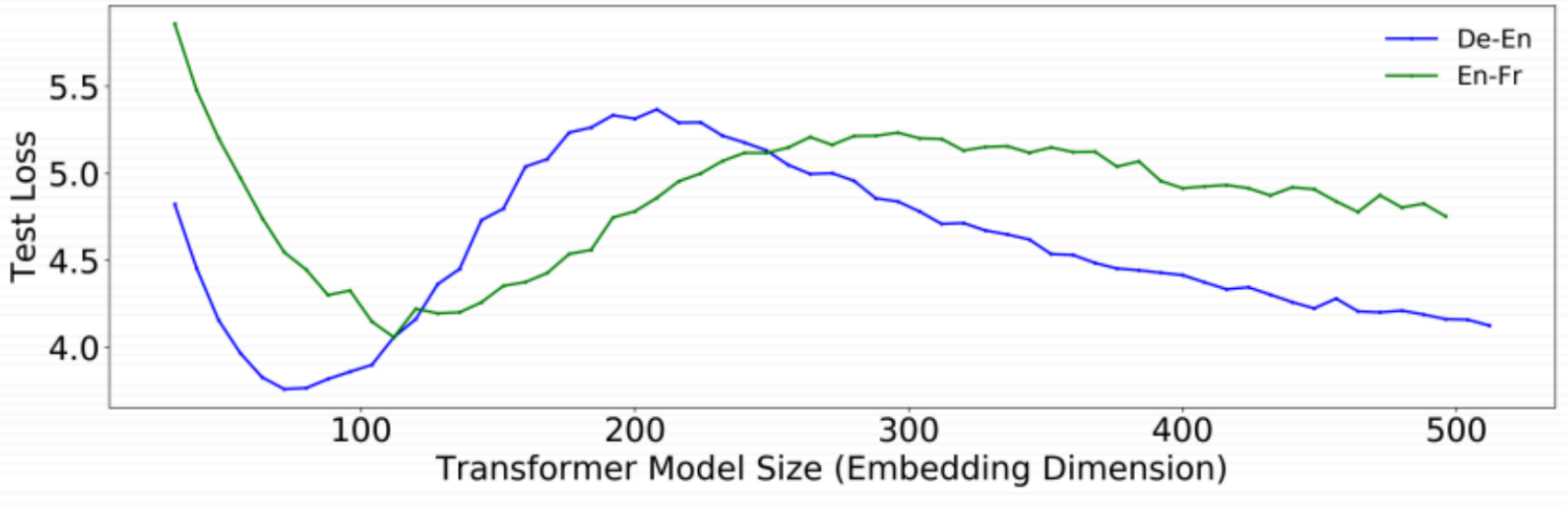
(a) **CIFAR-100.** There is a peak in test error even with no label noise.



(b) **CIFAR-10.** There is a “plateau” in test error around the interpolation point with no label noise, which develops into a peak for added label noise.

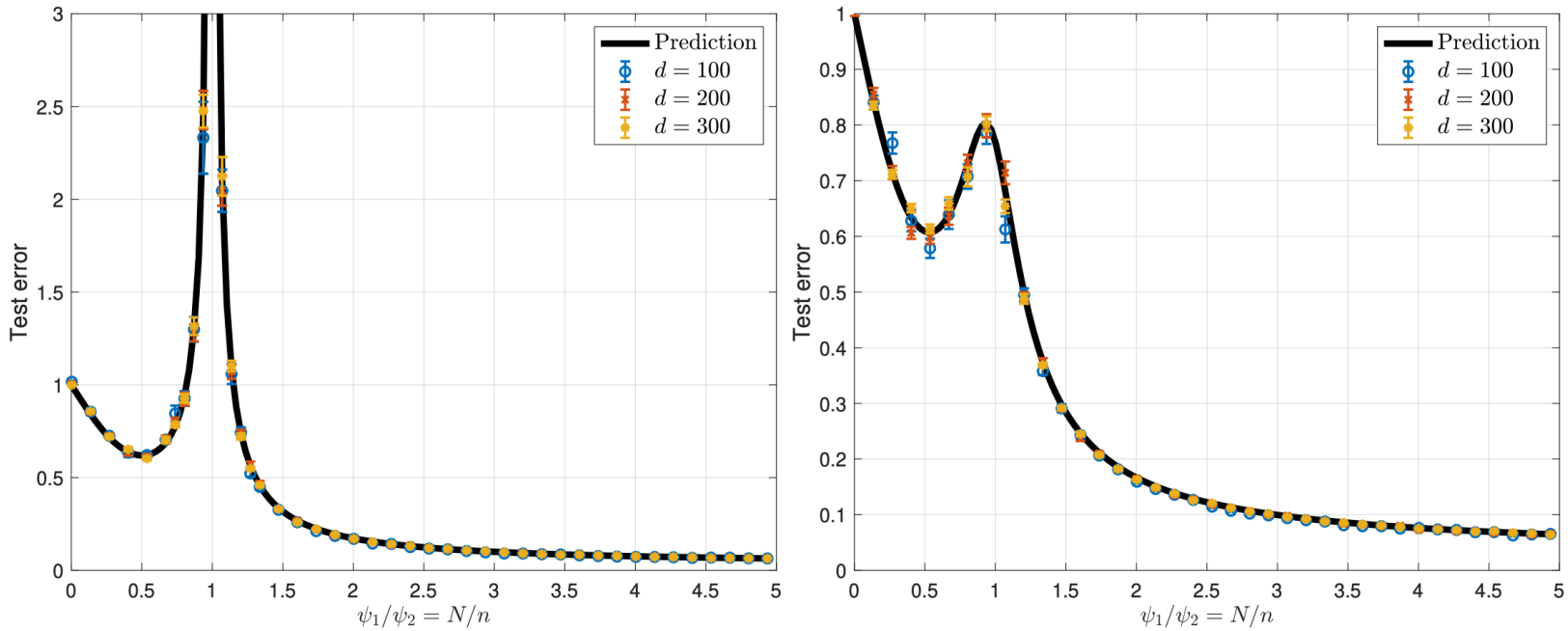
Double descent

Widespread phenomenon, across architectures (Nakkiran et al. ’19):



Double descent

Widespread phenomenon, also in kernels (can be formally proved in some concrete settings [Mei and Montanari ’20]), random forests, etc.



Double descent

Also in other quantities such as train time, dataset, etc (Nakkiran et al. ’19):

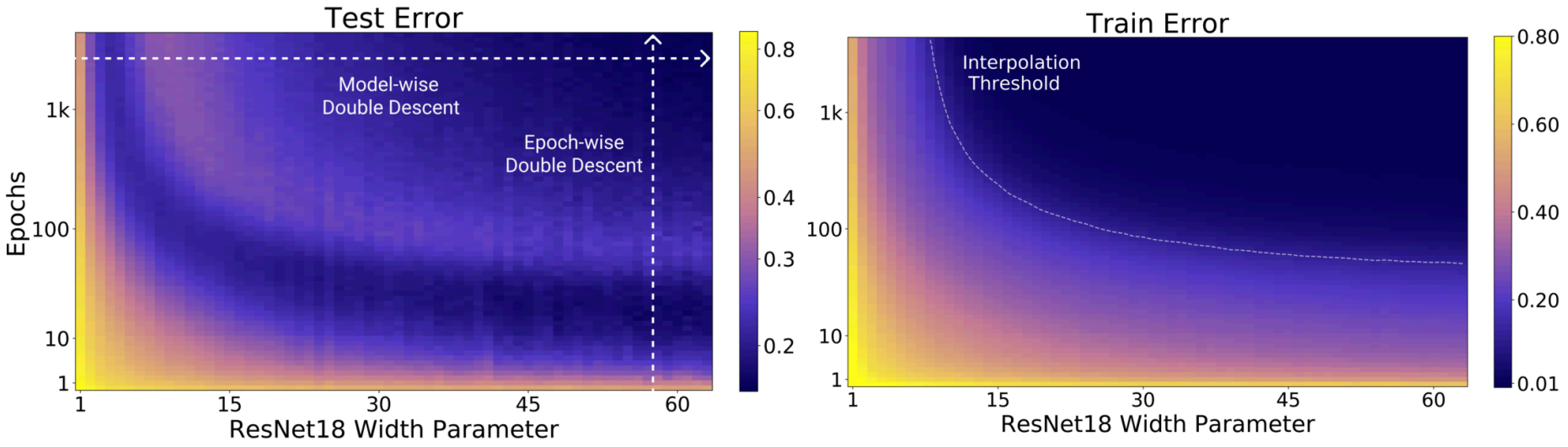
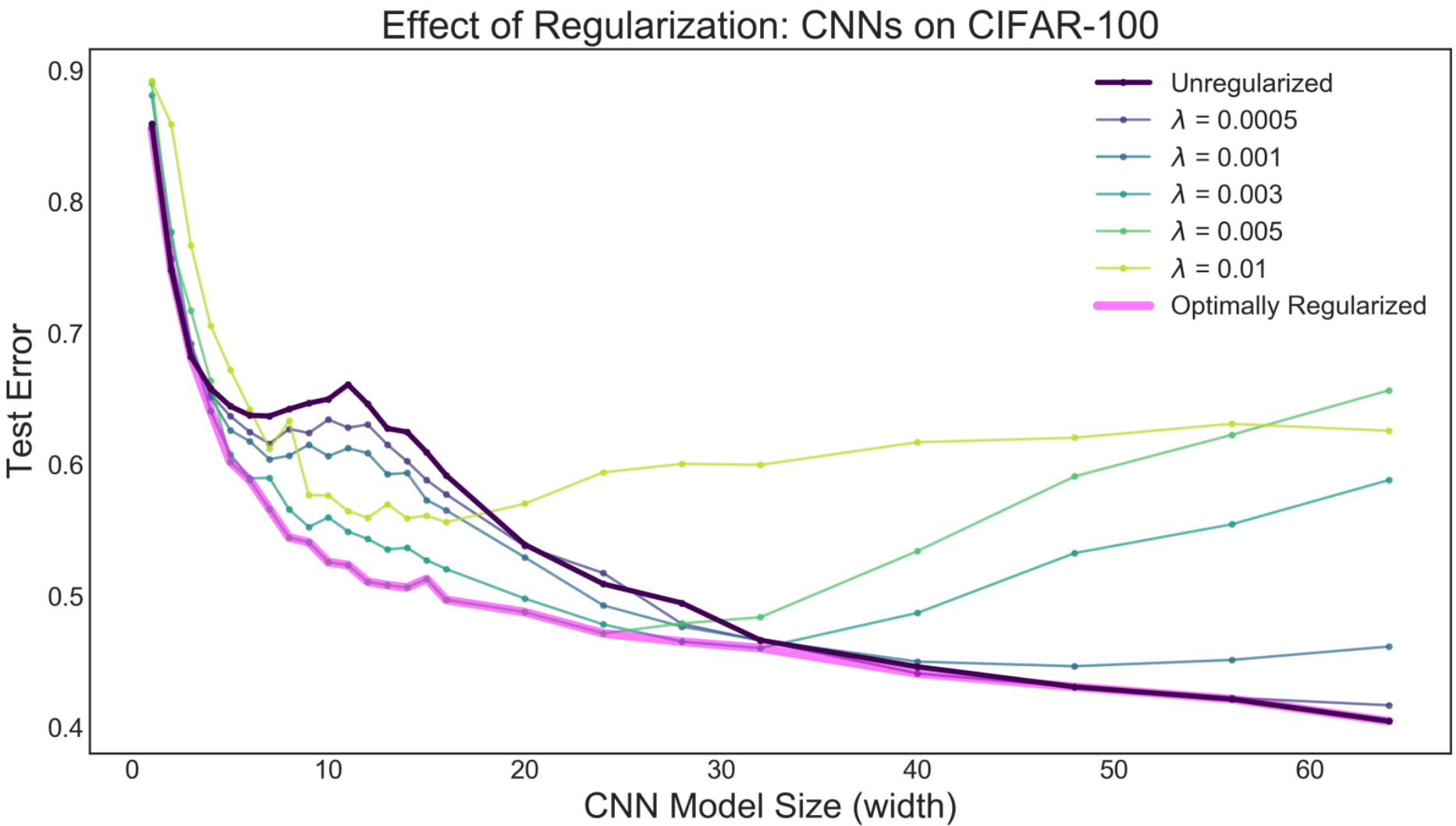


Figure 2: **Left:** Test error as a function of model size and train epochs. The horizontal line corresponds to model-wise double descent–varying model size while training for as long as possible. The vertical line corresponds to epoch-wise double descent, with test error undergoing double-descent as train time increases. **Right** Train error of the corresponding models. All models are Resnet18s trained on CIFAR-10 with 15% label noise, data-augmentation, and Adam for up to 4K epochs.

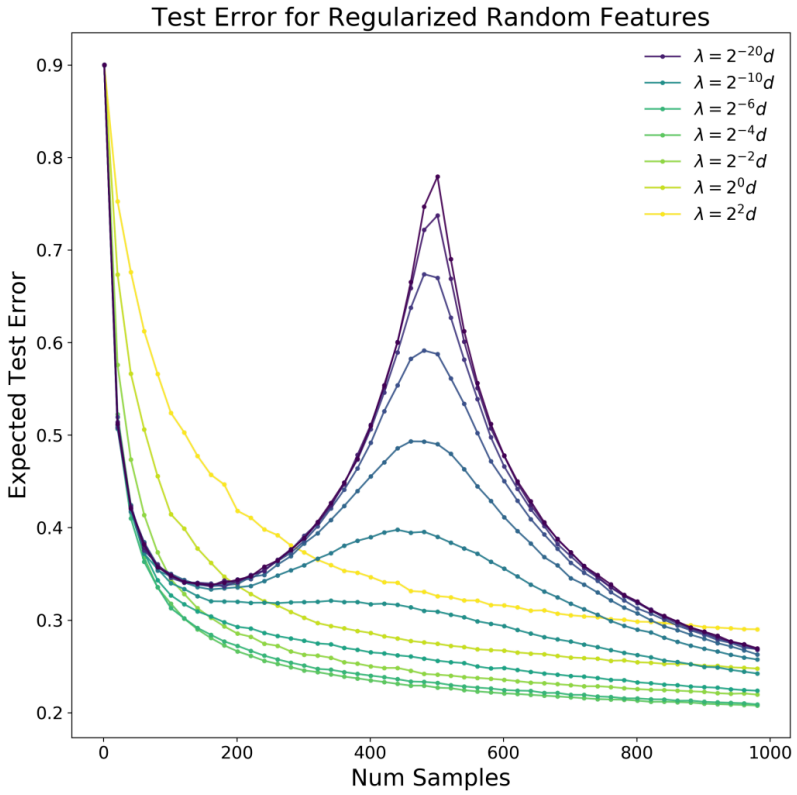
Double descent

Optimal regularization can mitigate double descent [Nakkiran et al. ’21]:

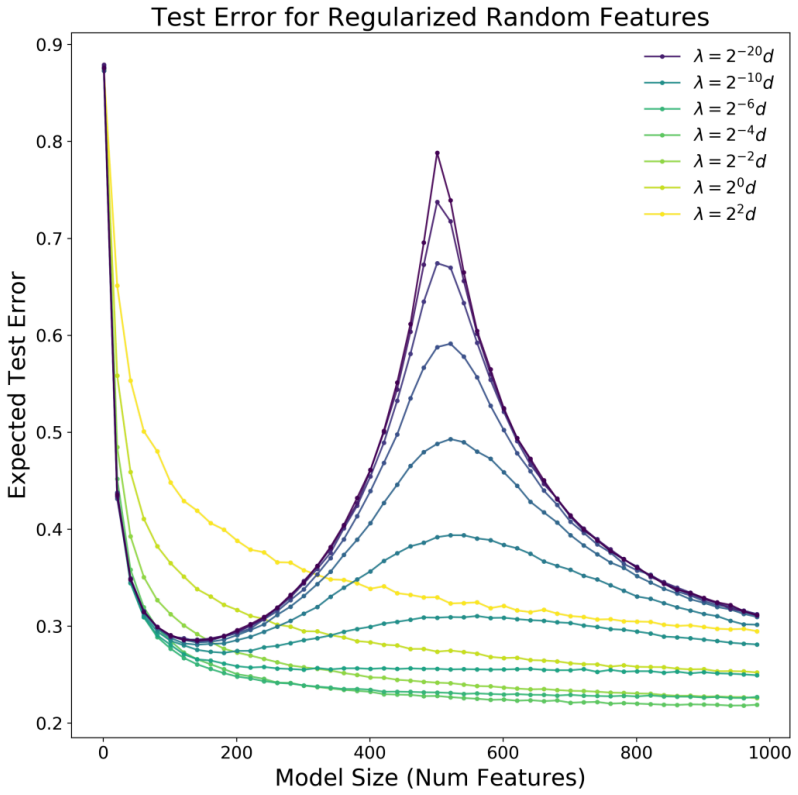


Double descent

Optimal regularization can mitigate double descent [Nakkiran et al. ’21]:



a) Test Classification Error vs. Number of Training Samples.



(b) Test Classification Error vs. Model Size (Number of Random Features).

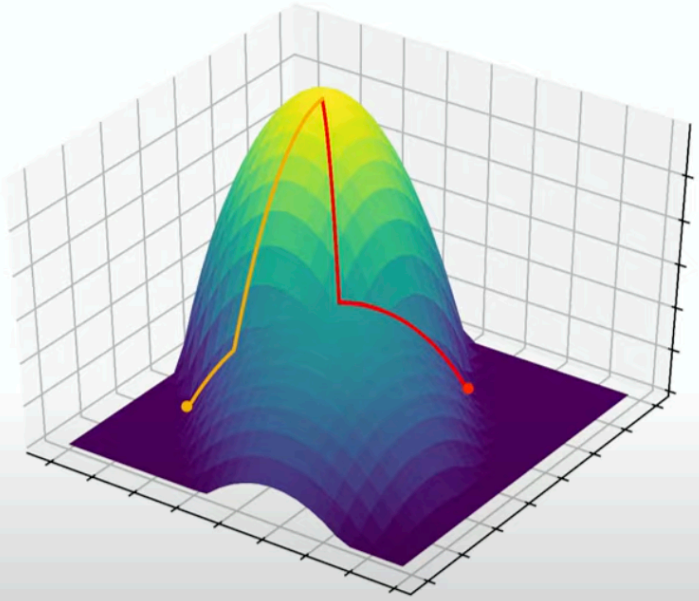
Implicit Regularization

Different optimization algorithm

➔ Different bias in optimum reached

➔ Different Inductive bias

➔ Different generalization properties



Implicit Bias

Margin:

- Linear predictors:
  - Gradient descent, mirror descent, natural gradient descent, steepest descent, etc maximize margins with respect to different norms.
- Non-linear:
  - Gradient descent maximizes margin for homogeneous neural networks.
  - Low-rank matrix sensing: gradient descent finds a low-rank solution.

Separation between NN and kernel

- For approximation and optimization, neural network has no advantage over kernel. Why NN gives better performance: generalization.

- [Allen-Zhu and Li '20] Construct a class of functions $\mathscr{F}$ such that $y = f(x)$ for some $f \in \mathscr{F}$:
  - no kernel is sample-efficient;
  - Exists a neural network that is sample-efficient.