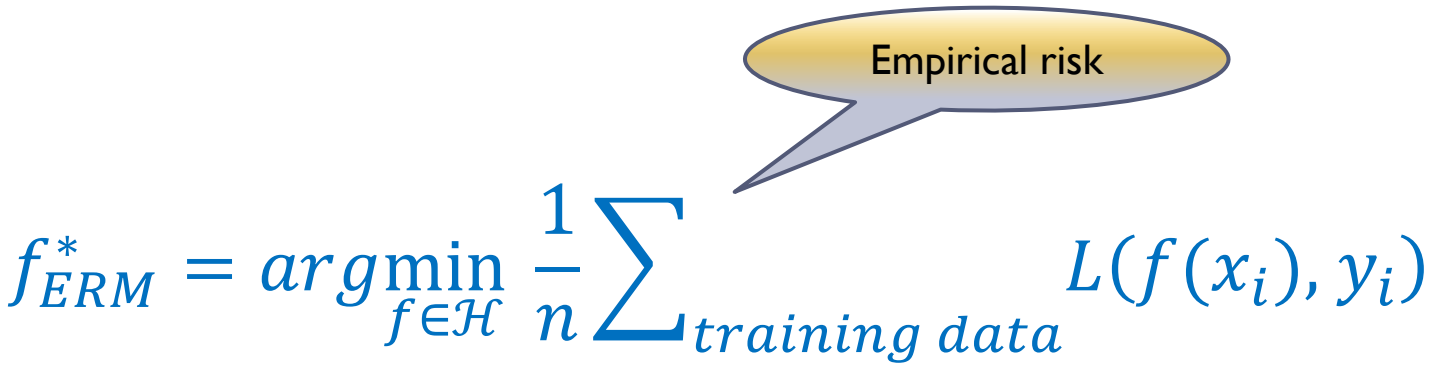
From Classical Statistics to Modern ML: the Lessons of Deep Learning

Mikhail Belkin
Ohio State University,
Department of Computer Science and Engineering,
Department of Statistics

IAS Workshop on Theory of Deep Learning: Where next?

Empirical Risk Minimization

Most theoretical analyses for ML are based on ERM:



Minimize empirical risk over a class of functions $\mathcal{H}$.

The ERM/SRM theory of learning

Goal of ML: $f^* = arg\min_f E_{unseen\ data}\, L(f(x), y)$

Goal of ERM: $f^*_{ERM} = arg\min_{f_w \in \mathcal{H}} \frac{1}{n} \sum_{training\ data} L(f_w(x_i), y_i)$

1. *The theory of induction is based on the* ***uniform law of large numbers.***
2. *Effective methods of inference must include* ***capacity control.***

V. Vapnik, Statistical Learning Theory, 1998

1. Empirical loss of any $f \in \mathcal{H}$ approximates expected loss of f.
2. $\mathcal{H}$ contains functions that approximate f^*.

(1)+(2) $\Rightarrow$ $E_{unseen\ data}\, L\left(f^*_{ERM}(x), y\right) \approx E_{unseen\ data}\, L(f^*(x), y)$

…

Uniform laws of large numbers

WYSIWYG bounds VC-dim, fat shattering, Rademacher, covering numbers, margin…

Model or function complexity, e.g., VC, margin or $\|f\|_{\mathcal{H}}$

Expected risk: what you get

Empirical risk: what you see

$$E(L(f^*_{ERM}, y)) \leq \frac{1}{n}\sum L\left(f^*_{ERM}(x_i), y_i\right) + O^*\left(\sqrt{\frac{c}{n}}\right)$$

Margin and other "a posteriori" bounds allow $\mathcal{H}$ to be data-dependent.

Capacity control

...

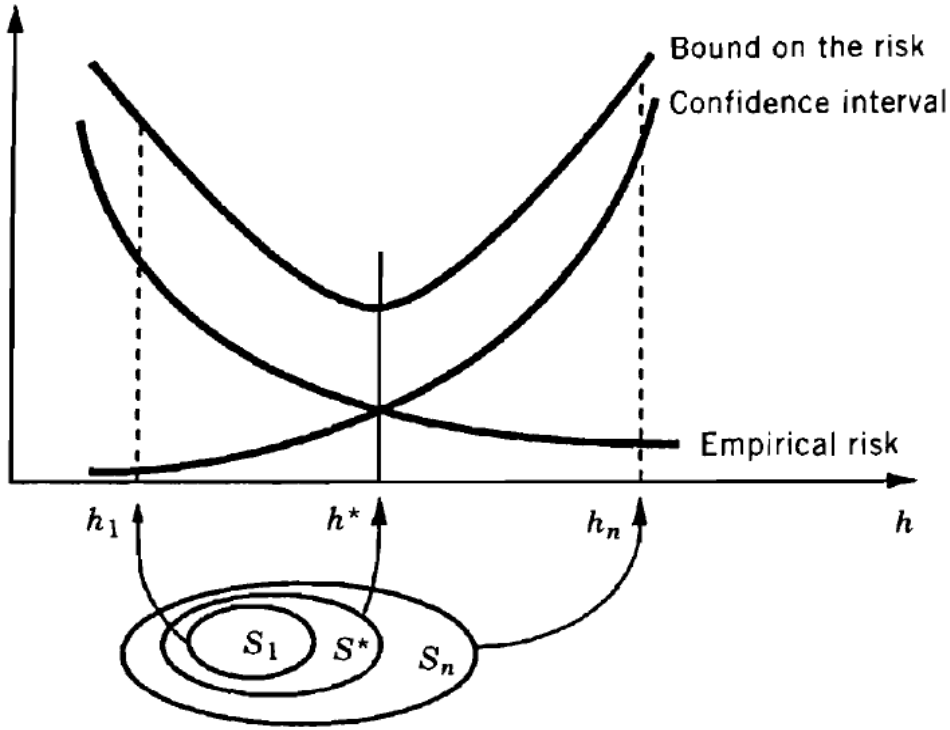
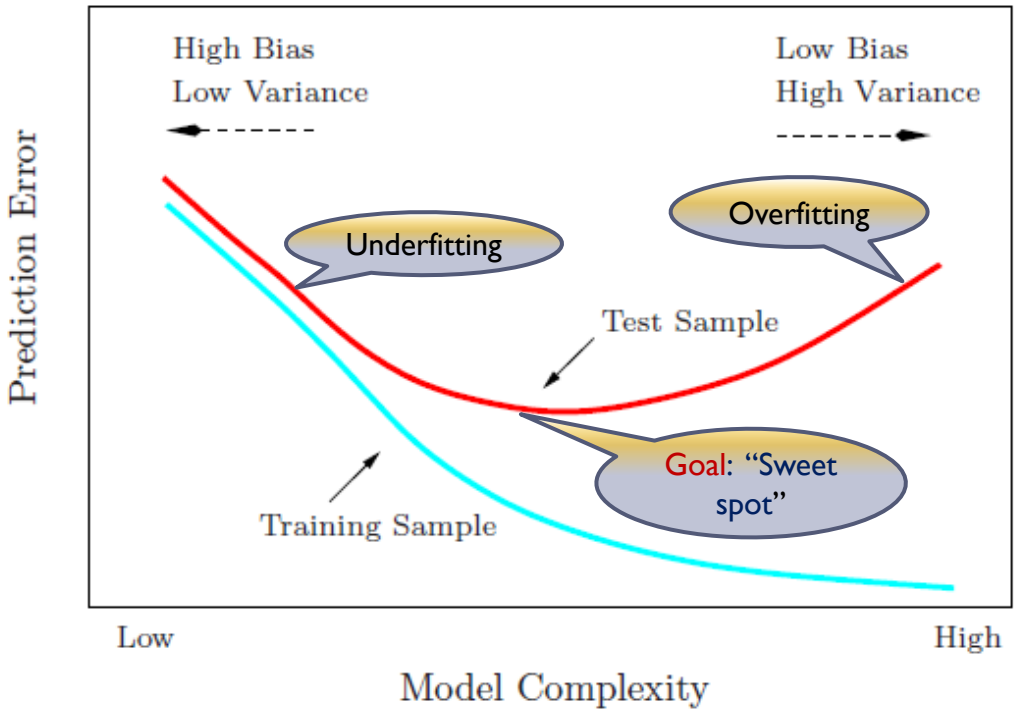
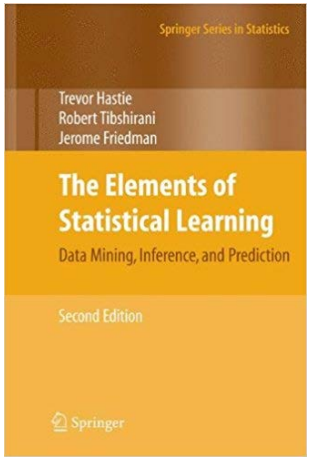


FIGURE 6.2. The bound on the risk is the sum of the empirical risk and of the confidence interval. The empirical risk is decreased with the index of element of the structure, while the confidence interval is increased. The smallest bound of the risk is achieved on some appropriate element of the structure.

V.Vapnik, **Statistical Learning Theory**, 1998

U-shaped generalization curve





However, a model with zero training error is overfit to the training data and will typically generalize poorly.

Interpolation

Page 194

Does interpolation overfit?

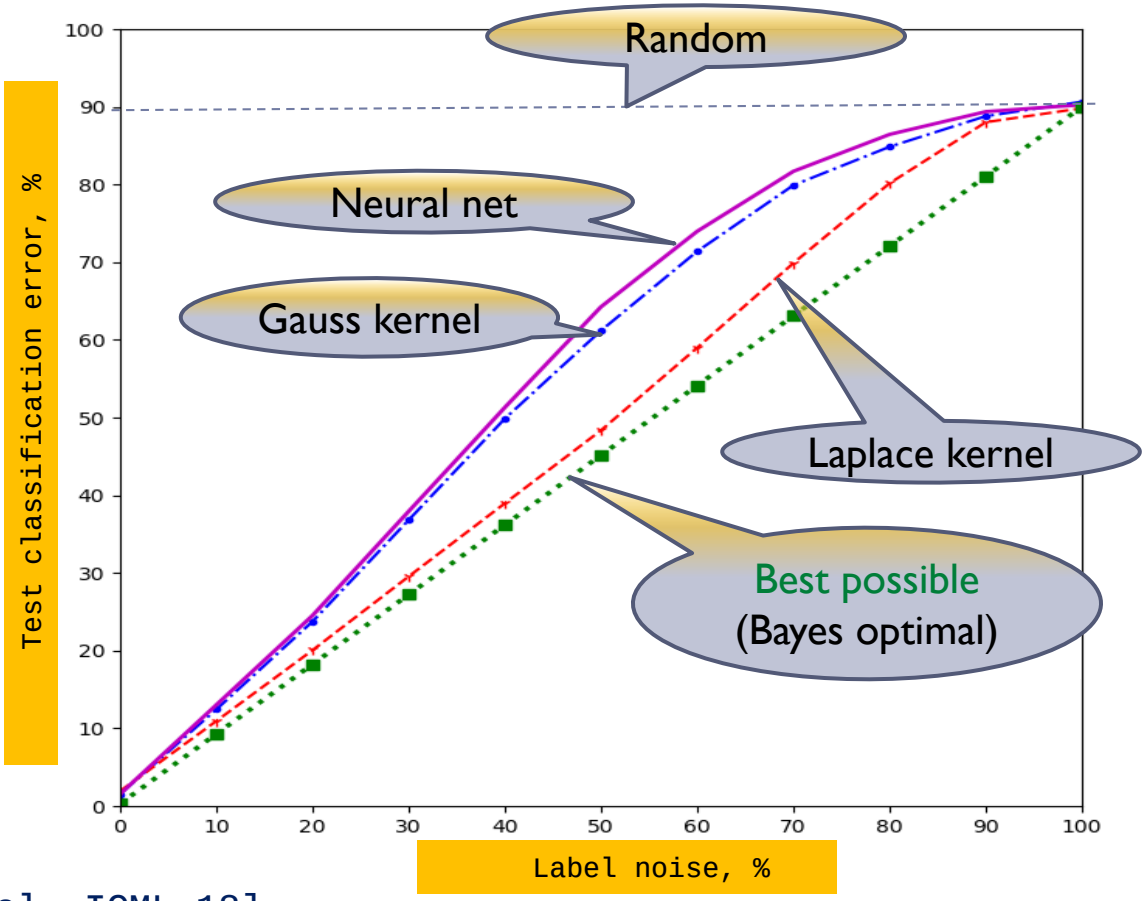
model	# params	random crop	weight decay	train accuracy	test accuracy
Inception	1,649,402	yes	yes	100.0	89.05
		yes	no	100.0	89.31
		no	yes	100.0	86.03
		no	no	100.0	85.75

[CIFAR 10, from *Understanding deep learning requires rethinking generalization*, Zhang, et al, 2017]

Suggestive, but does not on its own invalidate the ERM theory/uniform bounds.

Interpolation does not overfit even for very noisy data

All methods (except Bayes optimal) have zero training square loss.



[B., Ma, Mandal, ICML 18]

Uniform bounds:

VC-dim, fat shattering, Rademacher, covering numbers, PAC-Bayes, margin…

Model or function complexity, e.g., VC or $\|f\|_{\mathcal{H}}$

Test loss Training loss

$$E(L(f^*, y)) \leq \frac{1}{n}\sum L(f^*(x_i), y_i) + O^*\left(\sqrt{\frac{c}{n}}\right)$$

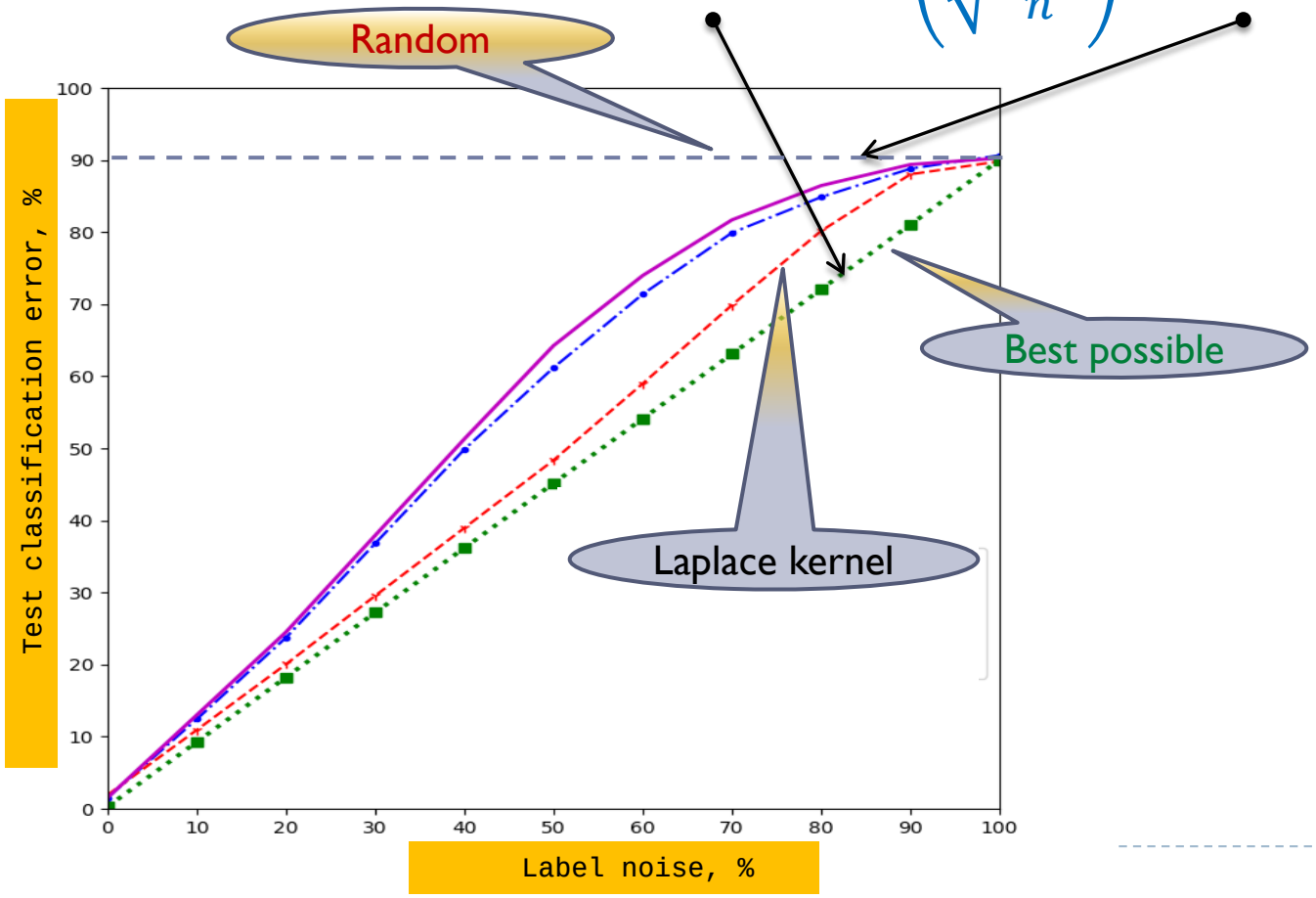
$= 0$

Can uniform bounds account for generalization under interpolation?

Bounds?

What kind of generalization bound could work here?

$$0.7 < O^*\left(\sqrt{\frac{c(n)}{n}}\right) < 0.9$$



Why bounds fail

$$\underset{\text{correct}}{0.7} < O^*\left(\sqrt{\frac{c(n)}{n}}\right) < \underset{\text{nontrivial}}{0.9} \qquad n \to \infty$$

1. The constant in O^* needs to be exact. There are no known bounds like that.

2. Conceptually, how would the quantity $c(n)$ "know" about the Bayes risk?

Interpolation is best practice for deep learning

From Ruslan Salakhutdinov's tutorial (Simons Institute, 2017):

The best way to solve the problem from practical standpoint is you build a very big system … basically you want to make sure you hit the zero training error.

Historical recognition

Yann Lecun (IPAM talk, 2018):

Deep learning breaks some basic rules of statistics.

Leo Breiman
Statistics Department, University of California, Berkeley, CA 94305;
e-mail: leo@stat.berkeley.edu

Written in 1995

Reflections After Refereeing Papers for NIPS

For instance, there are many important questions regarding neural networks which are largely unanswered. There seem to be conflicting stories regarding the following issues:

- Why don't heavily parameterized neural networks overfit the data?

Where we are now: the key lesson

The new theory of induction cannot be based on uniform laws of large numbers with capacity control.

Where next?

Generalization theory for interpolation?

What theoretical analyses do we have?

- VC-dimension/Rademacher complexity/covering/Pac-Bayes/margin bounds.
  - Cannot deal with interpolated classifiers when Bayes risk is non-zero.
  - Generalization gap cannot be bound when empirical risk is zero.
- Algorithmic stability.
  - Does not apply when empirical risk is zero, expected risk nonzero.
- Regularization-type analyses (Tikhonov, early stopping/SGD, etc.)
  - Diverge as $\lambda \to 0$ for fixed n.
- Classical smoothing methods (nearest neighbors, Nadaraya–Watson).
  - Most classical analyses do not support interpolation.
  - But 1-NN! (Also Hilbert regression Scheme, [Devroye, et al. 98])

Uniform bounds:

training loss $\approx$ expected loss

$= 0$

Typically Diverge

Oracle bounds

expected loss $\approx$ optimal loss

A way forward?

1-nearest neighbor classifier is very suggestive.

Interpolating classifier with a non-trivial (sharp!) performance guarantee.

Twice the Bayes risk [Cover, Hart, 67].

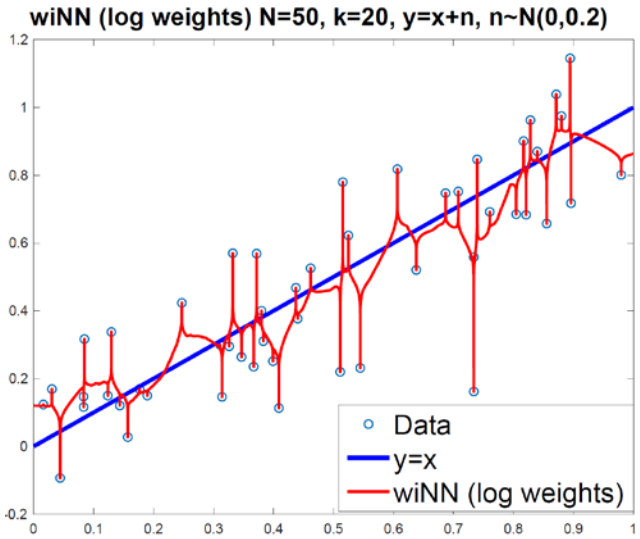
- Analysis not based on complexity bounds.
- Estimating expected loss, not the generalization gap.

Interpolated k-NN schemes

$$f(x) = \frac{\sum y_i k(x_i,x)}{\sum k(x_i,x)}$$

$$k(x_i,x) = \frac{1}{||x-x_i||^\alpha}, \quad k(x_i,x) = -\log||x-x_i||$$

(cf. Shepard's interpolation)



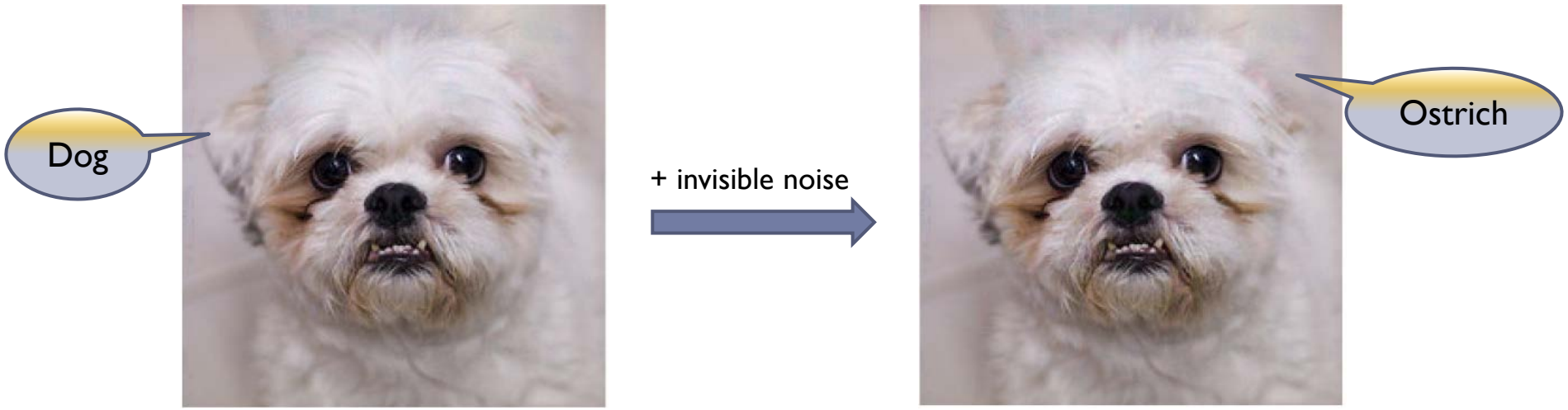
Theorem:

Weighted (interpolated) k-nn schemes with certain singular kernels are consistent (converge to Bayes optimal) for classification in any dimension.

Moreover, statistically (minimax) optimal for regression in any dimension.

[B., Hsu, Mitra, NeuriPS 18] [B., Rakhlin, Tsybakov, AIStats 19]

Interpolation and adversarial examples



From Szegedy, at al, ICLR 2014

Theorem: adversarial examples for interpolated classifiers are asymptotically dense (assuming the labels are not deterministic).

[B., Hsu, Mitra, NeuriPS 18]

This talk so far:

A. Effectiveness of interpolation.
B. Theory of interpolation cannot be based on uniform bounds.
C. Statistical validity of interpolating nearest neighbor methods.

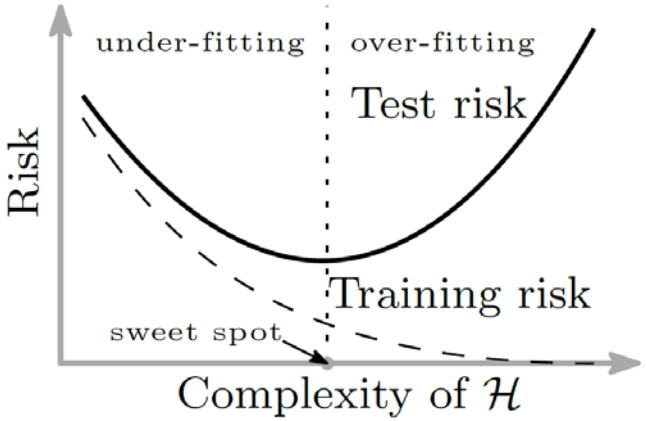
Yet, there is a mismatch between A and C. Methods we considered theoretically seem quite different from those used in practice.

Key questions:

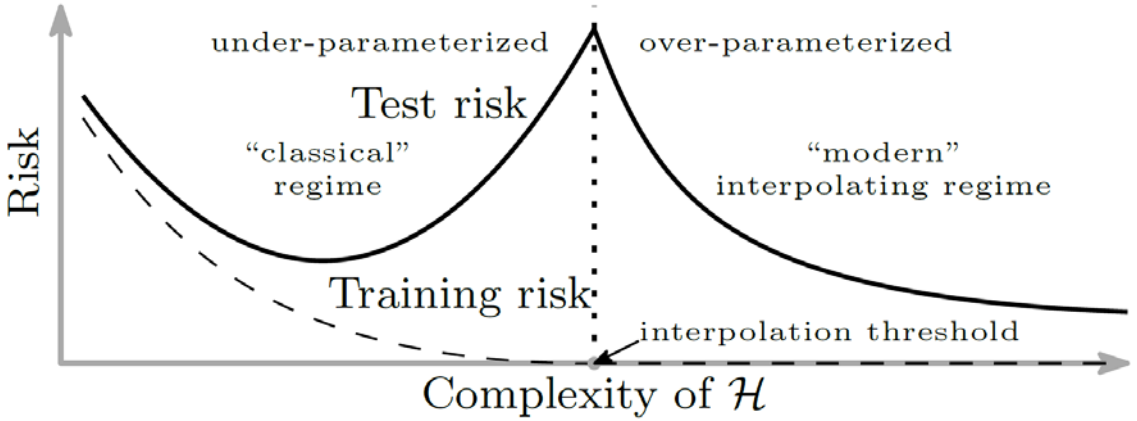
- How do classical analyses relate to interpolation?
- Dependence of generalization on model complexity?
- What is the role of optimization?

“Double descent” risk curve

Classical risk curve

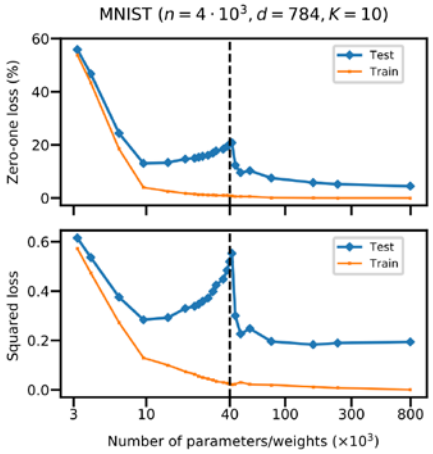


New “double descent” risk curve

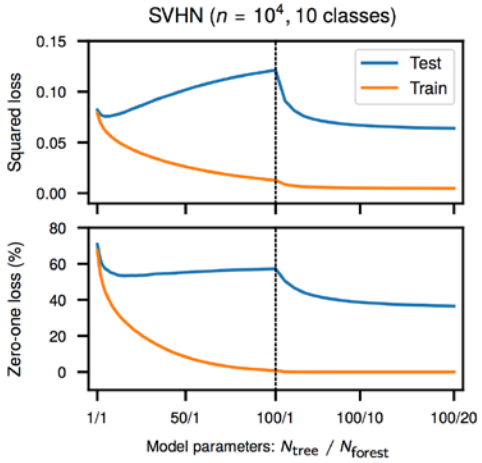


[B., Hsu, Ma, Mandal, PNAS 2019]

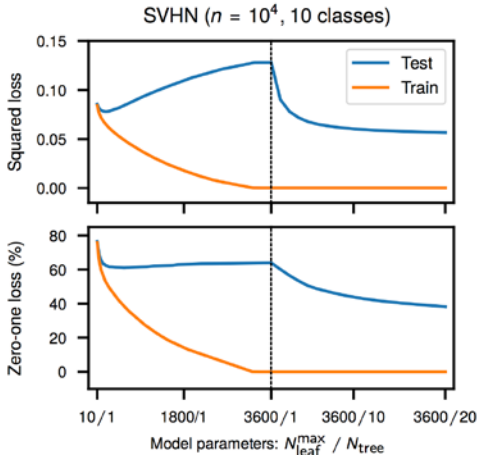
Fully connected network

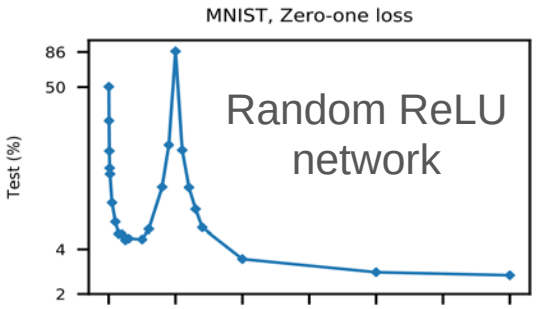


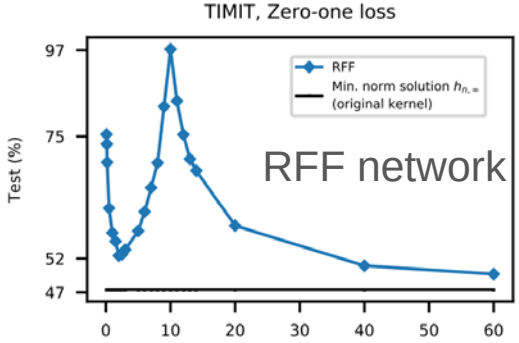
Random Forest

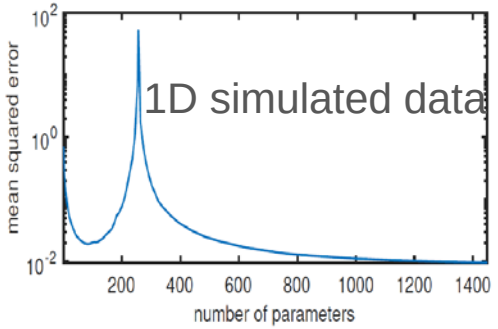


L2-boost

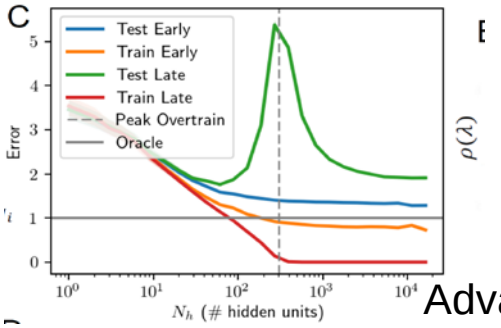




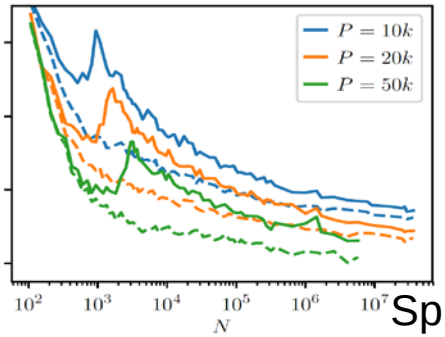




[B., Hsu, Ma, Mandal, 18]

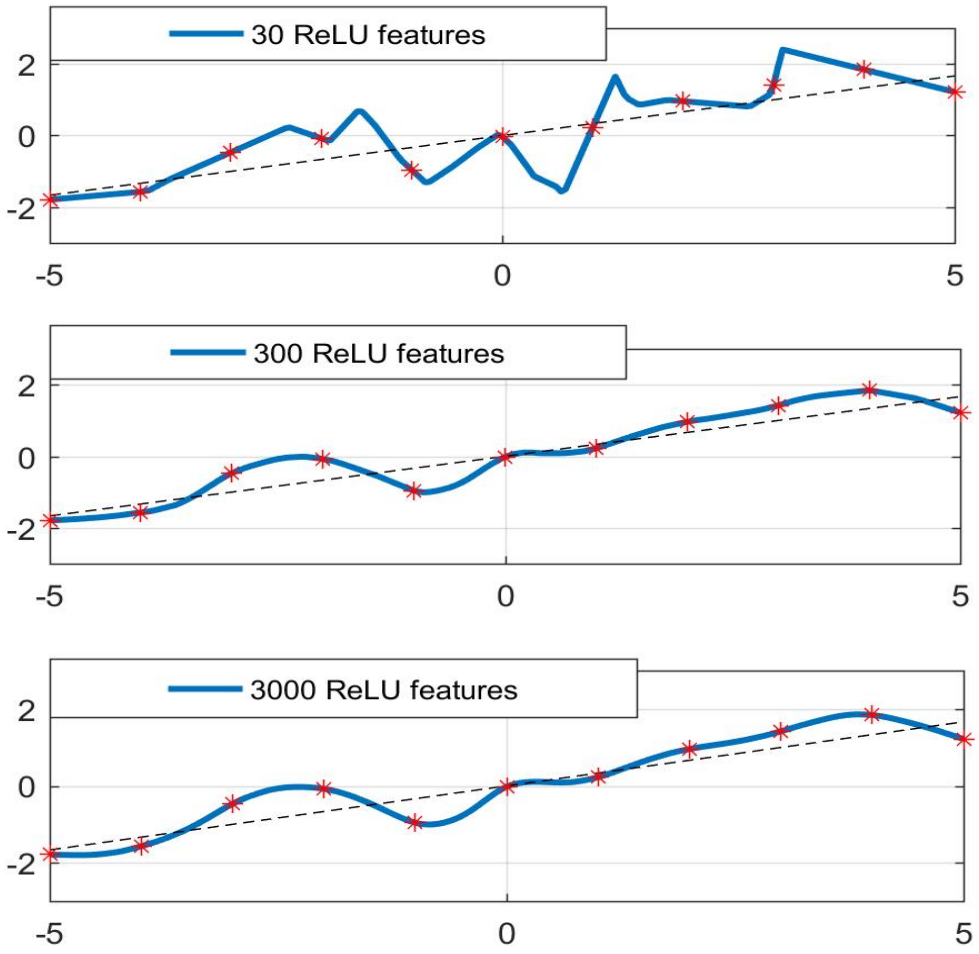


Advani, Saxe, 2017



Spigler, et al, 2018

More parameters are better: an example



Random Fourier networks

Random Fourier Features networks [Rahimi, Recht, NIPS 2007]

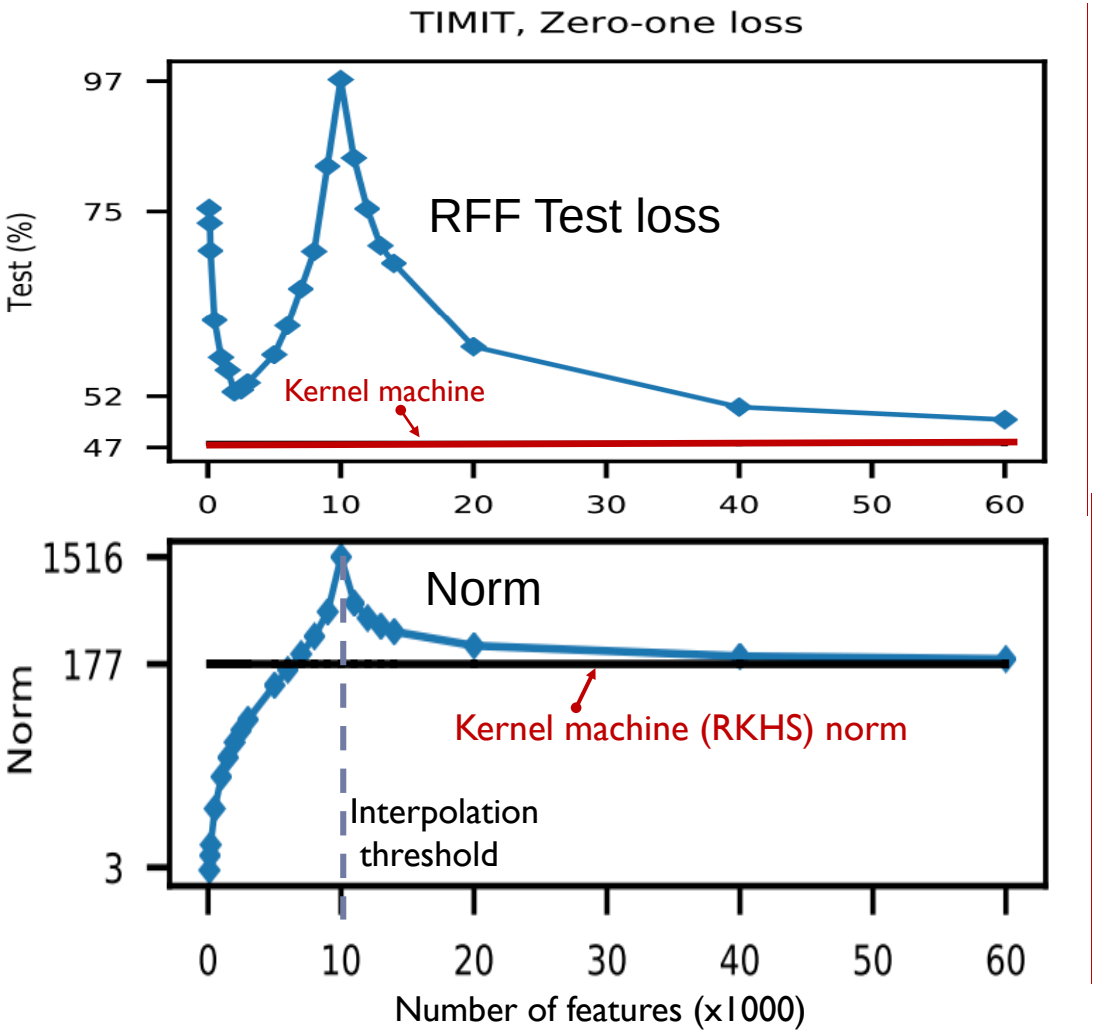
$$h_{n,N}(x) = \sum_{j=1}^{N} \alpha_j \, e^{i\pi\langle w_j, x\rangle}$$

Neural network with one hidden layer, cos non-linearity, fixed first layer weights. Hidden layer of size N. Data size n.

Key property:

$$\lim_{N\to\infty} h_{n,N}(x) = \text{kernel machine}$$

What is the mechanism?



$N \to \infty$ -- infinite neural net

=

kernel machine

=

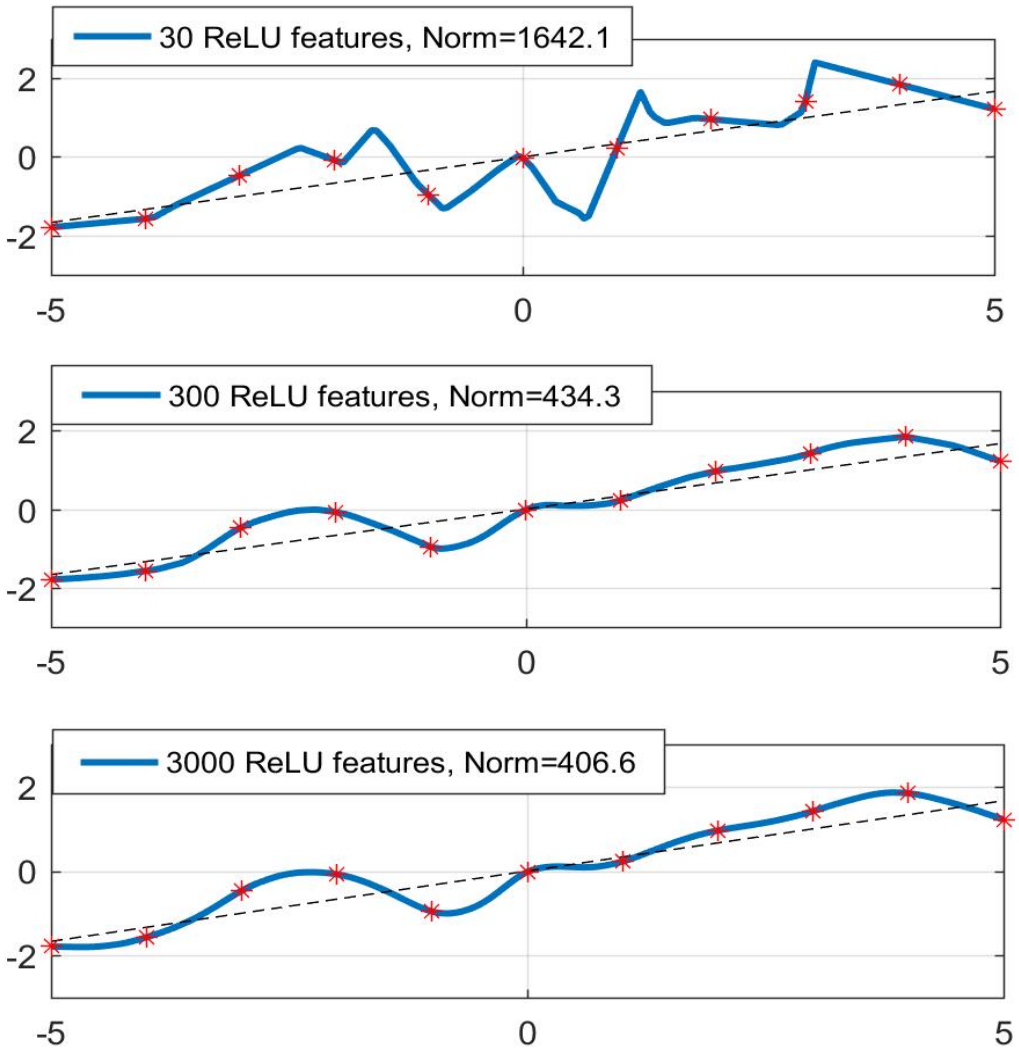
minimum norm solution

$argmin_{h \in \mathcal{H},\, h(x_i)=y_i} \; ||h||_{\mathcal{H}}$

More features ⇒

better approximation
to minimum norm solution

Smaller norm ensures smoothness



Is infinite width optimal?

Infinite net (kernel machine) $h_{n,\infty}$ is near-optimal empirically.

Suppose $\forall_i \;\; y_i = h^*(x_i)$ for some $h^* \in \mathcal{H}$ (Gaussian RKHS).

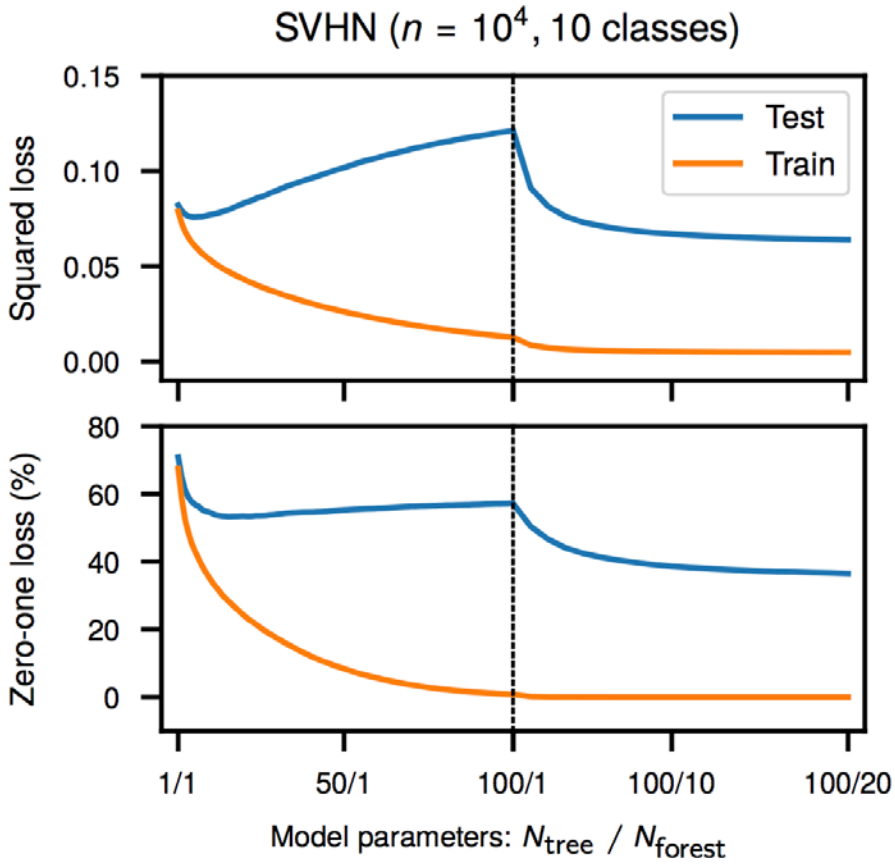
Theorem (noiseless case):

$$\left|h^*(x) - \; h_{n,\infty}(x)\right| < \; Ae^{-B(n/\log n)^{1/d}} \; ||h^*||_{\mathcal{H}}$$

Compare to $O\left(\frac{1}{\sqrt{n}}\right)$ for classical bias-variance analyses.

[B., Hsu, Ma, Mandal, PNAS 19]

Smoothness by averaging

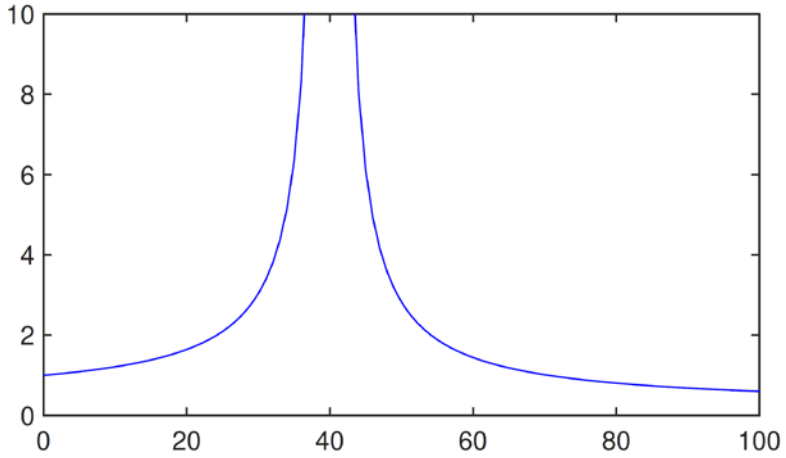


An average of interpolating trees is interpolating and better than any individual tree.

Cf. PERT [Cutler, Zhao 01]

Double Descent in Random Feature settings

Choosing maximum number of features is provably optimal under the "weak random feature" model.



[B., Hsu, Xu, 19].

Related work: [Bartlett, Long, Lugosi, Tsigler 19], [Hastie, Montanari, Rosset, Tibshirani 19] [Mitra, 19], [Muthukumar, Vodrahalli, Sahai, 19] [Mei, Montanari, 19] [Liang, Rakhlin, 19], [Liang, Rakhlin, Zhai, 19]

Significant evidence that deep neural networks exhibit similar properties.

Framework for modern ML

Occam's razor based on inductive bias:

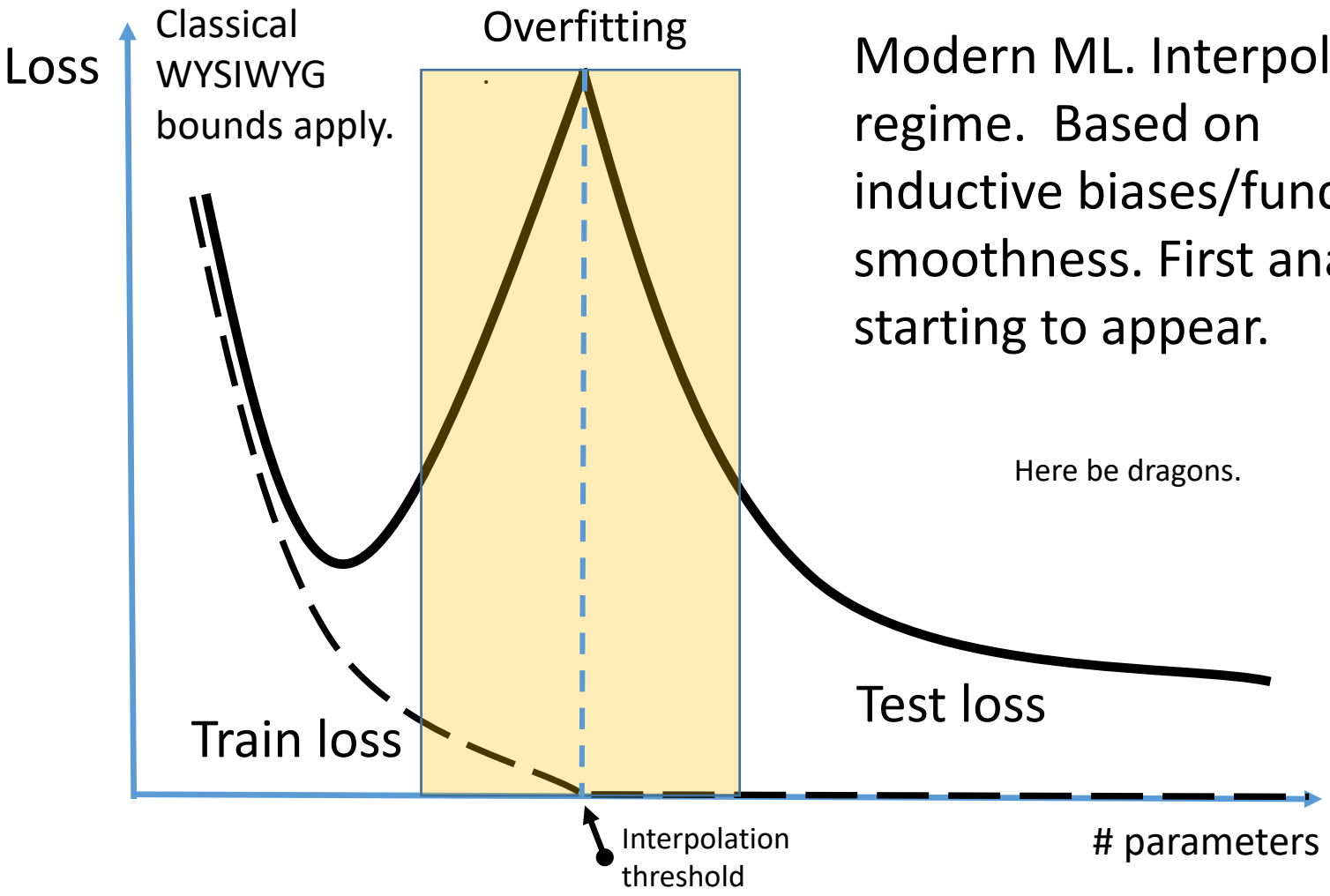
Maximize smoothness subject to interpolating the data.

Three ways to increase smoothness:

- Explicit: minimum functional norm solutions
  - Exact: kernel machines.
  - Approximate: RFF, ReLU features.
- Implicit: SGD/optimization (Neural networks)
- Averaging (Bagging, L2-boost).

All coincide for kernel machines.

The landscape of generalization



Modern ML. Interpolation regime. Based on inductive biases/functional smoothness. First analyses starting to appear.

Here be dragons.

This talk

- Statistical theory of interpolation.
  - Why classical bounds do not apply.
  - Statistical validity of interpolation.

- The generalization landscape of Machine Learning.
  - Double Descent: reconciling interpolation and the classical U curve.
  - Occams razor: more features is better.

- Interpolation and optimization
  - Easy optimization + fast SGD (+ good generalization).
  - Learning from deep learning for efficient kernel machines.

Optimization: classical

Classical (under-parametrized):

- Many local minima.
- SGD (fixed step size) does not converge.

Modern Optimization

Modern (interpolation/over-parametrized).

1. Every local minimum is global (for networks wide enough)

[Li, Ding, Sun, 18], [Yu, Chen, 95]

2. Local methods converge to global optima

[Kawaguchi, 16] [Soheil, et al, 16] [Bartlett, et al, 17] [Soltanolkotabi, et al, 17, 18] [Du, et al, 19] …

3. Small batch SGD (fixed step size) converges as fast as GD per iteration.

[Ma, Bassily, **B**., ICML 18] [Bassily, Ma, **B**., 18]

Why SGD?

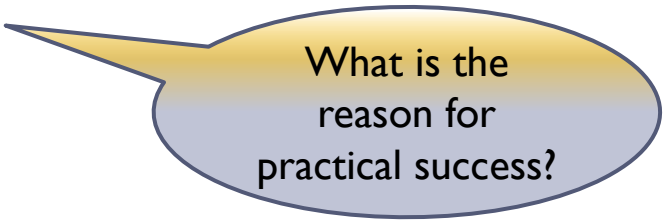
$$w^* = \operatorname*{argmin}_w L(w) = \operatorname*{argmin}_w \frac{1}{n} \sum L_i(w)$$

SGD Idea: optimize $\sum L_i(w)$, m at a time.

Error after t steps

GD: e^{-t}

SGD: $1/t$



Key point: SGD is not simply GD with noisy gradient estimates.

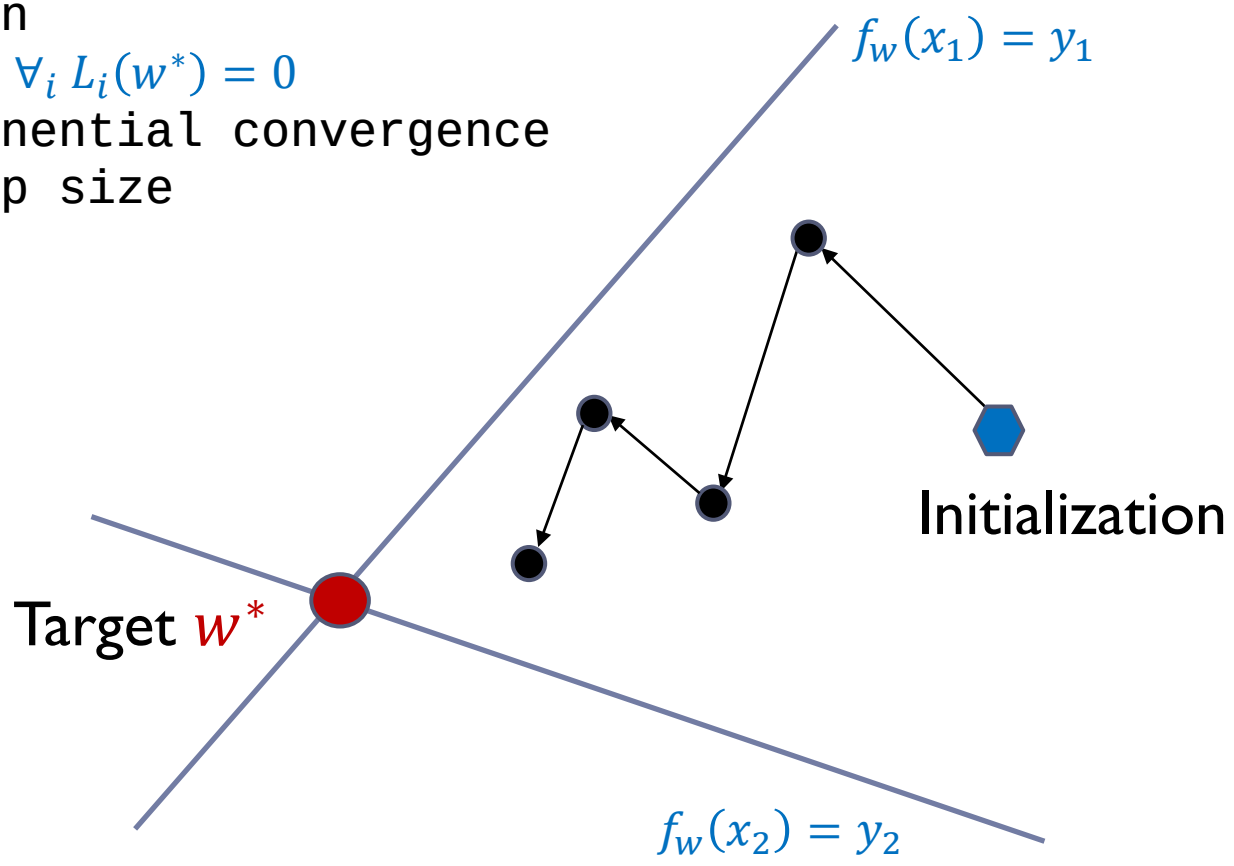
SGD under interpolation

Key observation:

Interpolation

$f_{w^*}(x_i) = y_i \;\Rightarrow\; \forall_i\, L_i(w^*) = 0$

implies exponential convergence w. fixed step size



SGD is (much) faster than GD

"Theorem": one SGD iteration with mini-batch size $m^* = \frac{tr\, H}{\lambda_1(H)}$ is equivalent to an iteration (=epoch) of full GD.

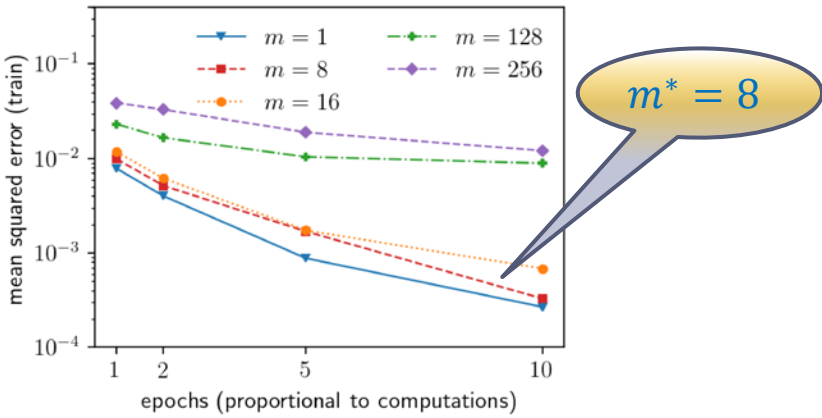
Savings per epoch: n/m^*.

Real data example.

One step of SGD with mini-batch $m^* \approx 8$

=

One step of GD.



[Ma, Bassily, **B.**, ICML 18]

The power of interpolation

Optimization in modern deep learning:

- overparametrization
- interpolation
- fast SGD
- GPU

SGD computational gain over GD $O\left(\frac{n}{m^*}\right)$

* GPU ~ 100x over CPU.

$n = 10^6, m^* = 8$:

SGD on GPU $\sim 10^7$x faster than GD on CPU!

Learning from deep learning: fast and effective kernel machines

EigenPro 2.0

Dataset	Size	Dimension	Our method (GPU)	ThunderSVM (GPU) [WSL+18]	LibSVM (CPU)
TIMIT	$1 \cdot 10^5$	440	**15 s**	480 s	1.6 h
SVHN	$7 \cdot 10^4$	1024	**13 s**	142 s	3.8 h
MNIST	$6 \cdot 10^4$	784	**6 s**	31 s	9 m
CIFAR-10	$5 \cdot 10^4$	1024	**8 s**	121 s	3.4 h

EigenPro: preconditioned SGD for kernel machines. Batch size/preconditioner optimized to take full advantage of GPU.

Code: https://github.com/EigenPro

[Ma, B., NIPS 17, SysML 19]

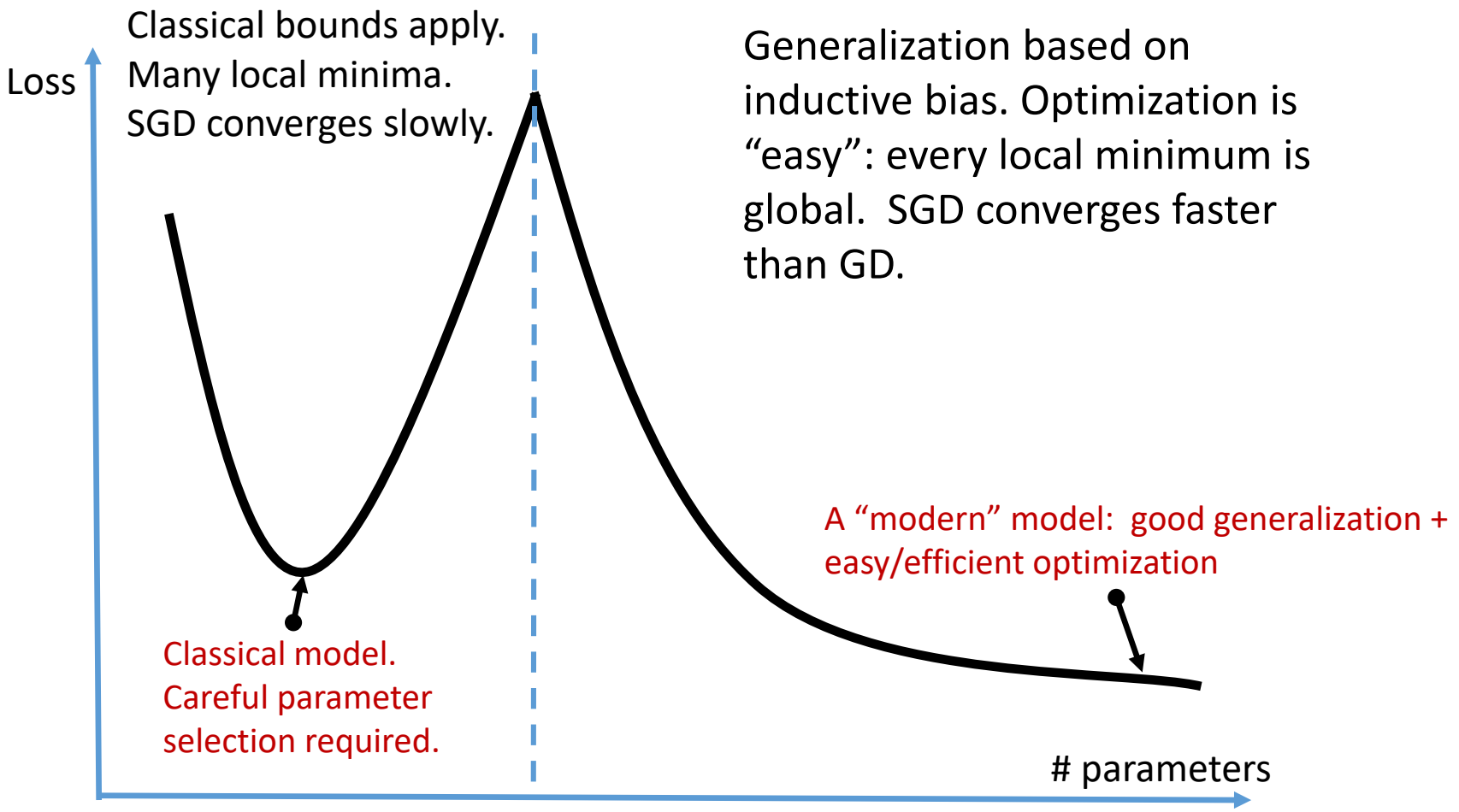
Points and lessons

- ERM cannot a sole foundation for modern ML.
  - Instead of uniform laws of large numbers, need to study inductive biases. Still early but analyses are starting to appear.
  - Key concept is interpolation, not over-parametrization. Over-parametrization enables interpolation but is not sufficient. Classical methods, kernels machines/splines are infinitely over-parametrized.
- Empirical loss is a useful optimization target, not a meaningful statistic for the expected loss.
- Optimization is qualitatively different under interpolation.
  - Every local minimum is global.
  - SGD is overwhelmingly faster than GD.

From classical statistics to modern ML

Classical.

Modern ML (interpolation regime).



Collaborators:

Siyuan Ma, Ohio State University
Soumik Mandal, Ohio State University

Daniel Hsu, Columbia University
Raef Bassily, Ohio State University
Partha Mitra, Spring Harbor Labs.
Sasha Rakhlin, MIT
Sasha Tsybakov, ENSAE

Thank you