Generalization of Deep Learning

Yuan YAO

HKUST

Some Theories are limited but help:

- *Approximation Theory and Harmonic Analysis* : **What functions are represented well by deep neural networks, without suffering the curse of dimensionality and better than shallow networks?**
  - Sparse (local), hierarchical (multiscale), compositional functions avoid the curse dimensionality
  - Group (translation, rotational, scaling, deformation) invariances achieved as depth grows
- *Generalization:* **How can deep learning generalize well without overfitting the noise?**
  - Double descent curve with overparametrized models
  - Implicit regularization of SGD: Max-Margin classifier
  - "Benign overfitting"?
- *Optimization*: **What is the landscape of the empirical risk and how to optimize it efficiently?**
  - Wide networks may have simple landscape for GD/SGD algorithms …

Empirical Risk vs. Population Risk

- Consider the ***empirical risk*** *minimization* under **i.i.d.** (independent and identically distributed) samples

$$\hat{R}_n(\theta) = \hat{\mathbb{E}}_n \ell(y, f(x;\theta)) := \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i;\theta)) + \mathcal{R}_n(\theta)$$

- The ***population risk*** with respect to unknown distribution

$$R(\theta) = \mathbb{E}_{(x,y)\sim P} \ell(y, f(x;\theta))$$

Optimization vs. Generalization

- Fundamental Theorem of Machine Learning (for 0-1 misclassification loss, called 'errors' below)

$$\underbrace{R(\theta)}_{\text{test/validation/generalization loss}} = \underbrace{\hat{R}_n(\theta)}_{\text{training loss}} + \underbrace{R(\theta) - \hat{R}_n(\theta)}_{\text{generalization gap}}$$

$$\sup_{\theta \in \Theta} |R(\theta) - \hat{R}_n(\theta)| \leq Complexity(\Theta)$$

e.g. Rademacher complexity

- How to make training loss/error small? – Optimization issue
- How to make generalization gap small? – Model Complexity issue

Uniform Convergence: Another View

- For $\theta^* \in \arg\min_{\theta \in \Theta} R(\theta)$ and $\widehat{\theta}_n \in \arg\min_{\theta \in \Theta} \hat{R}_n(\theta)$,

$$\begin{aligned}
\underbrace{R(\widehat{\theta}_n) - R(\theta^*)}_{\text{excess risk}} \quad = \quad & \underbrace{R(\widehat{\theta}_n) - \hat{R}_n(\widehat{\theta}_n)}_{\text{A}} + \dots \\
& + \underbrace{(\hat{R}_n(\widehat{\theta}_n) - \hat{R}_n(\theta^*))}_{\leq 0} + \dots \\
& + \underbrace{(\hat{R}_n(\theta^*) - R(\theta^*))}_{\text{B}}
\end{aligned}$$

- To make both A and B small,

$$\sup_{\theta \in \Theta} |R(\theta) - \hat{R}_n(\theta)| \leq Complexity(\Theta)$$

e.g. Rademacher complexity

Example: regression and square loss

- Given an estimate $\hat{f}$ and a set of predictors X, we can predict Y using

$$\hat{Y} = \hat{f}(X),$$

- Assume for a moment that both $\hat{f}$ and X are fixed. In regression setting,

$$\begin{aligned}\mathbb{E}(Y - \hat{Y})^2 &= \mathbb{E}[f(X) + \epsilon - \hat{f}(X)]^2 \\ &= \underbrace{[f(X) - \hat{f}(X)]^2}_{\textbf{Reducible}} + \underbrace{Var(\epsilon)}_{\textbf{Irreducible}} \quad , \end{aligned} \tag{2}$$

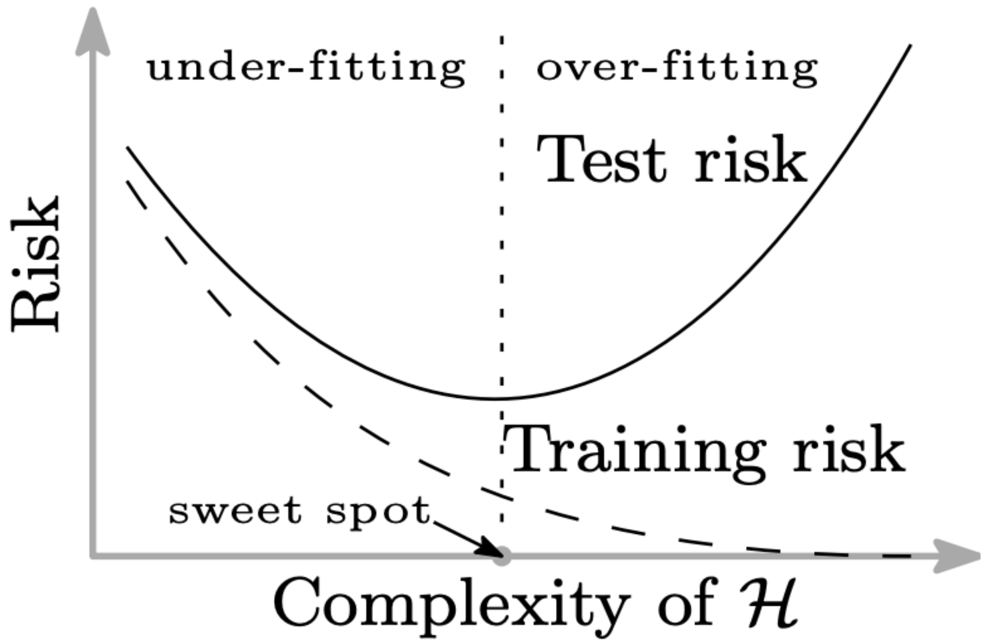
where $\mathbb{E}(Y - \hat{Y})^2$ represents the expected squared error between the predicted and actual value of Y, and $Var(\epsilon)$ represents the variance associated with the error term ϵ. An optimal estimate is to minimize the reducible error.

Bias-Variance Decomposition

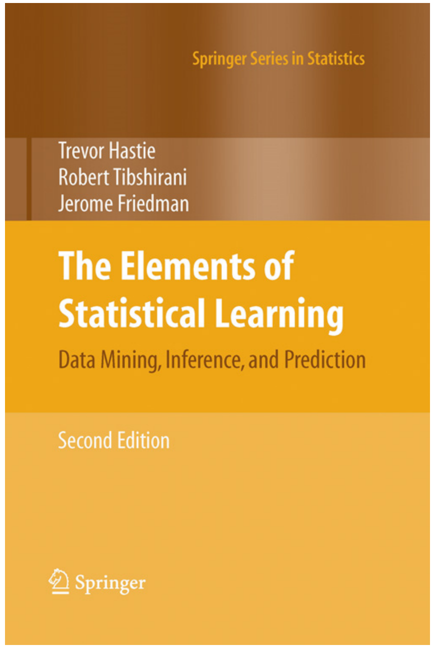
- Let $f(X)$ be the true function which we aim at estimating from a training data set $\mathcal{D}$.
- Let $\hat{f}(X;\mathcal{D})$ be the estimated function from the training data set $\mathcal{D}$.
- Take the expectation with respect to $\mathcal{D}$,

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{D}}\left[f(X)-\hat{f}(X;\mathcal{D})\right]^2 \\
&=\underbrace{\left[f(X)-\mathbb{E}_{\mathcal{D}}(\hat{f}(X;\mathcal{D}))\right]^2}_{\mathit{Bias}^2}+\underbrace{\mathbb{E}_{\mathcal{D}}\left[\left[\mathbb{E}_{\mathcal{D}}(\hat{f}(X;\mathcal{D}))-\hat{f}(X;\mathcal{D})\right]^2\right]}_{\mathit{Variance}}
\end{aligned}
$$

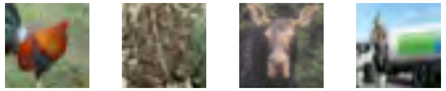
Bias-Variance Tradeoff



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



Why big models in NN generalize well?



CIFAR10

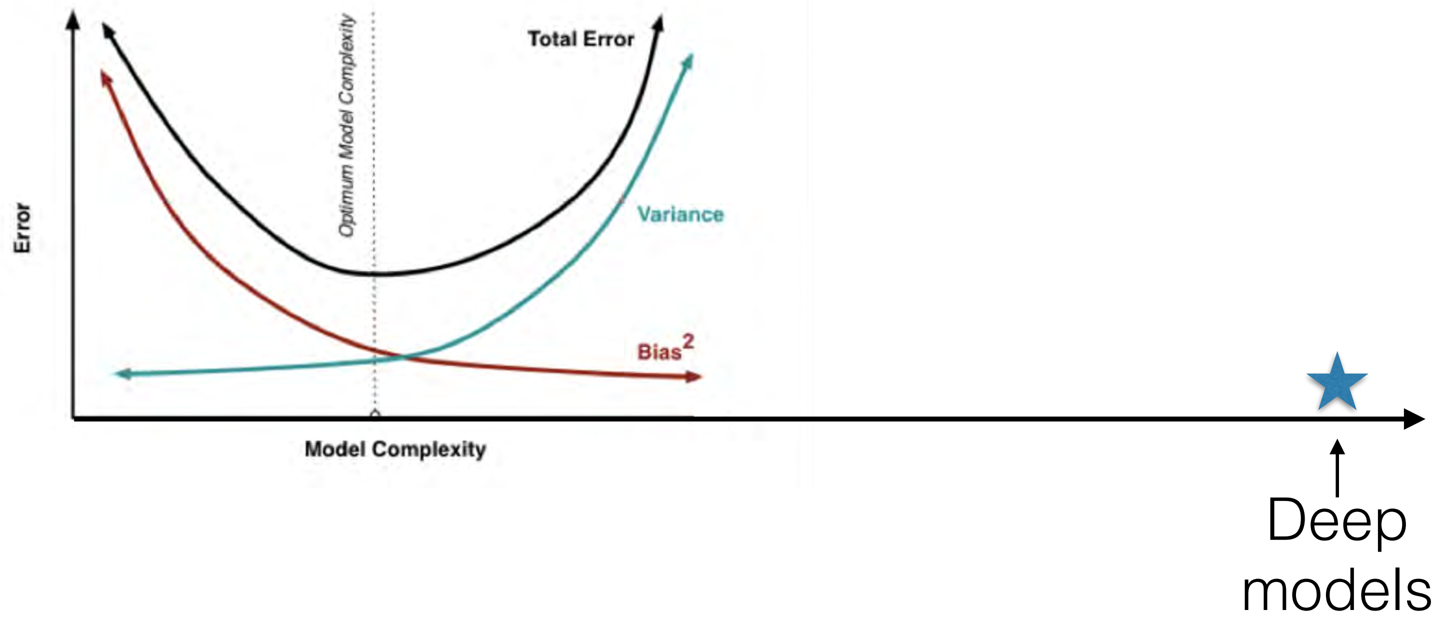
n=50,000
d=3,072
k=10

What happens when I turn off the regularizers?

Model	parameters	p/n	Train ***loss***	Test ***error***
CudaConvNet	145,578	2.9	0	23%
CudaConvNet (with regularization)	145,578	2.9	0.34	18%
MicroInception	1,649,402	33	0	14%
ResNet	2,401,440	48	0	13%

Chiyuan Zhang et al. 2016

The Bias-Variance Tradeoff?



Models where p>20n are common

Increasing # parameters

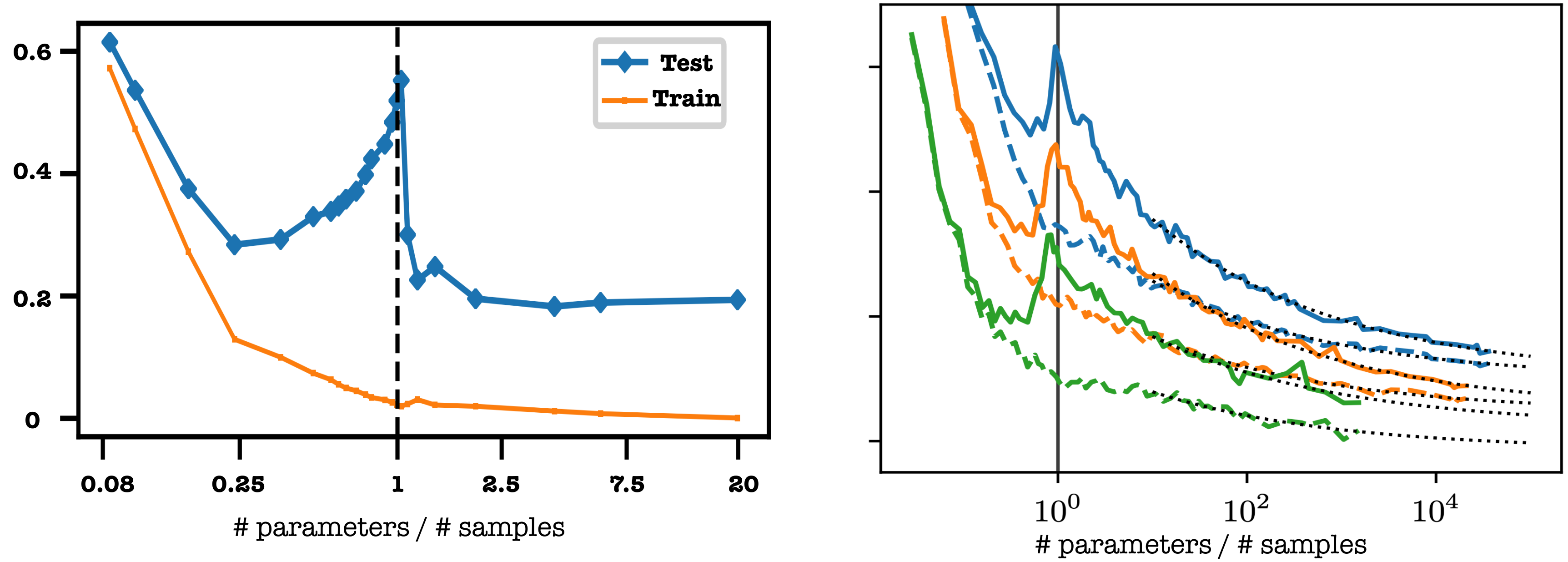
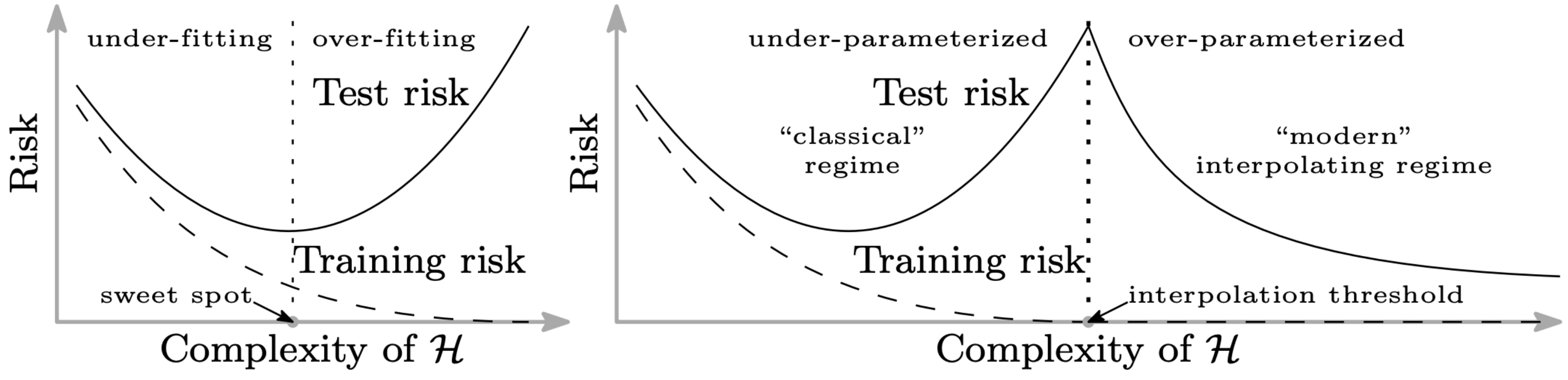


Figure: Experiments on MNIST. Left: [Belkin, Hsu, Ma, Mandal, 2018]. Right: [Spigler, Geiger, Ascoli, Sagun, Biroli, Wyart, 2018].

Similar phenomenon appeared in the literature [LeCun, Kanter, and Solla, 1991], [Krogh and Hertz, 1992], [Opper and Kinzel, 1995], [Neyshabur, Tomioka, Srebro, 2014], [Advani and Saxe, 2017].

"Double Descent"



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

(b) "double descent" risk curve

Figure: A cartoon by [Belkin, Hsu, Ma, Mandal, 2018].

- ✓ Peak at the interpolation threshold.
- ✓ Monotone decreasing in the overparameterized regime.
- ✓ Global minimum when the number of parameters is infinity.

Complementary rather than Contradiction

U-shaped curve

Test error vs model complexity that tightly controls generalization.

Examples: ℓ_2 norm in linear model, "k" in k nearest-neighbors.

Double-descent

Test error vs number of parameters.

Examples: # parameters in NN.

In NN, # parameters $\neq$ model complexity that tightly controls generalization.

[Bartlett, 1997], [Bartlett and Mendelson, 2002]

Let's go to two talks

- Prof. Misha Belkin (OSU/UCSD)
  - From Classical Statistics to Modern Machine Learning at Simons Institute at Berkeley
  - How interpolation models do not overfit…
- Prof. Song Mei (UC Berkeley)
  - Generalization of linearized neural networks: staircase decay and double descent, at HKUST
  - How simple linearized single-hidden-layer models help understand…

Thank you!