Discussion on generalization and margins

Daniel Hsu
Columbia University

August 5, 2020
JSM session on "Theoretical Advances In Deep Learning"

What is generalization in ~~machine~~ deep learning?

At least two definitions for *generalization error* are floated in the community:

What is generalization in ~~machine~~ deep learning?

At least two definitions for *generalization error* are floated in the community:

1. Out-of-sample (test) error rate

$$\mathrm{err}(f)$$

2. Difference between out-of-sample (test) and in-sample (training) error rates

$$\mathrm{err}(f) - \mathrm{err}(f; S_n)$$

What is generalization in ~~machine~~ deep learning?

At least two definitions for *generalization error* are floated in the community:

1. Out-of-sample (test) error rate

$$\mathrm{err}(f)$$

2. Difference between out-of-sample (test) and in-sample (training) error rates

$$\mathrm{err}(f) - \mathrm{err}(f; S_n)$$

We care about the former, empirical process theory is good for the latter (since $S_n \sim (\mathrm{Pr}_{\mathbf{x},y})^n$; "uniform convergence bounds")

What is generalization in ~~machine~~ deep learning?

At least two definitions for *generalization error* are floated in the community:

1. Out-of-sample (test) error rate

$$\mathrm{err}(f)$$

2. Difference between out-of-sample (test) and in-sample (training) error rates

$$\mathrm{err}(f) - \mathrm{err}(f; S_n)$$

We care about the former, empirical process theory is good for the latter (since $S_n \sim (\mathrm{Pr}_{\mathbf{x},y})^n$; "uniform convergence bounds")

Major use case: Analysis of Empirical Risk Minimization (ERM)

$$\min_{f \in \mathcal{F}} \ \mathrm{err}(f; S_n)$$

Consistency of models that perfectly fit training data

[Belkin, H., Mitra, 2018]: "Weighted & Interpolating k_n-NN" classifier $f_n \equiv f_{S_n}$ satisfies

$$\mathbb{E}_{S_n}\left[\Pr_{\mathbf{x}}\left(f_n(\mathbf{x}) \neq f_{\text{bayes}}(\mathbf{x})\right)\right] \to 0 \quad \text{as } n \to \infty$$

under regularity conditions on distribution of $\mathbf{x}$

Consistency of models that perfectly fit training data

[Belkin, H., Mitra, 2018]: "Weighted & Interpolating k_n-NN" classifier $f_n \equiv f_{S_n}$ satisfies

$$\mathbb{E}_{S_n}\left[\Pr_{\mathbf{x}}\left(f_n(\mathbf{x}) \neq f_{\text{bayes}}(\mathbf{x})\right)\right] \to 0 \quad \text{as } n \to \infty$$

under regularity conditions on distribution of $\mathbf{x}$

- In particular,

  $$\operatorname{err}(f_n; S_n) = 0 \quad \text{(always)}$$

  and

  $$\begin{aligned}\mathbb{E}_{S_n}\left[\left|\operatorname{err}(f_n) - \operatorname{err}(f_n; S_n)\right|\right] &= \mathbb{E}_{S_n}\left[\operatorname{err}(f_n)\right] \\ &\to \operatorname{err}(f_{\text{bayes}}).\end{aligned}$$

Consistency of models that perfectly fit training data

[Belkin, H., Mitra, 2018]: "Weighted & Interpolating k_n-NN" classifier $f_n \equiv f_{S_n}$ satisfies

$$\mathbb{E}_{S_n}\left[\Pr_{\mathbf{x}}\left(f_n(\mathbf{x}) \neq f_{\text{bayes}}(\mathbf{x})\right)\right] \to 0 \quad \text{as } n \to \infty$$

under regularity conditions on distribution of $\mathbf{x}$

- In particular,

$$\operatorname{err}(f_n; S_n) = 0 \quad \text{(always)}$$

  and

$$\begin{aligned}\mathbb{E}_{S_n}\left[\left|\operatorname{err}(f_n) - \operatorname{err}(f_n; S_n)\right|\right] &= \mathbb{E}_{S_n}\left[\operatorname{err}(f_n)\right] \\ &\to \operatorname{err}(f_{\text{bayes}}).\end{aligned}$$

- $\therefore$ Any uniform convergence bound that applies to f_n must "stall" at the Bayes error rate (which may be non-zero).

Consistency of models that perfectly fit training data

[Belkin, H., Mitra, 2018]: "Weighted & Interpolating k_n-NN" classifier $f_n \equiv f_{S_n}$ satisfies

$$\mathbb{E}_{S_n}\left[\Pr_{\mathbf{x}}\left(f_n(\mathbf{x}) \neq f_{\text{bayes}}(\mathbf{x})\right)\right] \to 0 \quad \text{as } n \to \infty$$

under regularity conditions on distribution of $\mathbf{x}$

- In particular,

$$\operatorname{err}(f_n; S_n) = 0 \quad \text{(always)}$$

  and

$$\begin{aligned}\mathbb{E}_{S_n}\left[\left|\operatorname{err}(f_n) - \operatorname{err}(f_n; S_n)\right|\right] &= \mathbb{E}_{S_n}\left[\operatorname{err}(f_n)\right] \\ &\to \operatorname{err}(f_{\text{bayes}}).\end{aligned}$$

- $\therefore$ Any uniform convergence bound that applies to f_n must "stall" at the Bayes error rate (which may be non-zero).
- (Similar results for squared-error regression.)

Uniform convergence and perfect-fit classifiers

Are there issues when $\mathrm{err}(f_{\mathrm{bayes}}) \approx 0$?

Uniform convergence and perfect-fit classifiers

Are there issues when $\mathrm{err}(f_{\mathrm{bayes}}) \approx 0$?

Theisen, Klusowski, Mahoney: Compelling MNIST analysis

- Many classifiers with $\mathrm{err}(f, S_n) = 0$ have low $\mathrm{err}(f)$
- There are classifiers with $\mathrm{err}(f, S_n) = 0$ and high $\mathrm{err}(f)$
- $\therefore$ "Uniform convergence bounds" still have problems ...

Uniform convergence and perfect-fit classifiers

Are there issues when $\mathrm{err}(f_{\mathrm{bayes}}) \approx 0$?

Theisen, Klusowski, Mahoney: Compelling MNIST analysis

- Many classifiers with $\mathrm{err}(f, S_n) = 0$ have low $\mathrm{err}(f)$
- There are classifiers with $\mathrm{err}(f, S_n) = 0$ and high $\mathrm{err}(f)$
- $\therefore$ "Uniform convergence bounds" still have problems ...

Possible fix: Only consider large margin classifiers (or other quantitative inductive bias)

- Schapire, Freund, Bartlett, and Lee (1998); Zhang (2002); ...
- But *a posteriori* bounds don't directly analyze the inductive bias achieved by the fitted model
- Sharp constrast with analyses of **Ji and Telgarsky**; **Ji and Telgarsky**; **Liang, Rakhlin, Zhai**; **Liang and Sur**; ...

BTW

PAC-Bayes approach to margin bounds (e.g., Langford and Shawe-Taylor, 2002) is a relevant bridge between worst-case and average-case analysis.

- Relevance: Maybe practitioners don't pick a (consistent) classifier at random

BTW

PAC-Bayes approach to margin bounds (e.g., Langford and Shawe-Taylor, 2002) is a relevant bridge between worst-case and average-case analysis.

- Relevance: Maybe practitioners don't pick a (consistent) classifier at random
- (But still has same issues as other *a posteriori* bounds.)

Support vector machines (SVMs)

Figure 1: Relevance

Support vector machines (SVMs)

Figure 1: Relevance

Vapnik (1979): mathematical definition of maximum margin linear classifier, along with a theory of generalization.

$$\begin{aligned} \min_{w \in \mathbb{R}^d} \quad & \|w\|_2 \\ \text{s.t.} \quad & y_i x_i^\mathsf{T} w \geq 1, \quad i = 1, \ldots, n. \end{aligned}$$

(All $y_i \in \{-1, 1\}$.)

Support vector machines (SVMs)

Figure 1: Relevance

Vapnik (1979): mathematical definition of maximum margin linear classifier, along with a theory of generalization.

$$\begin{aligned} \min_{w \in \mathbb{R}^d} \quad & \|w\|_2 \\ \text{s.t.} \quad & y_i x_i^{\mathsf{T}} w \geq 1, \quad i = 1, \ldots, n. \end{aligned}$$

(All $y_i \in \{-1, 1\}$.)

- Why not $\min_{w \in \mathbb{R}^d} \|w\|_2$ s.t. $x_i^{\mathsf{T}} w = y_i$ (interpolation)?

SVMs vs interpolation [Muthukumar *et al*, 2020]

$$K(x_1, x_2) = \sum_{k \geq 0} \frac{\sin(kx_1)\sin(kx_2) + \cos(kx_1)\cos(kx_2)}{(k+1)^{2m}}$$

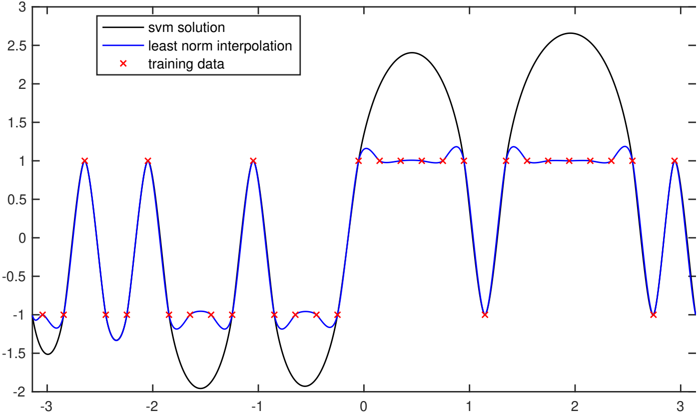

Figure 2: SVM solution vs least norm interpolation $(m = 1.5)$

Margins in very high-dimensions

- Toy setup similar to that of **Theisen, Klusowski, Mahoney**

$$
\begin{aligned}
(\mathbf{x}_i, y_i) &\sim_{\text{iid}} \frac{1}{2}(N_+, 1) + \frac{1}{2}(N_-, -1) \quad i = 1, \ldots, n \\
N_+ &= \mathcal{N}(\mu, I_d) \\
N_- &= \mathcal{N}(-\mu, I_d)
\end{aligned}
$$

Margins in very high-dimensions

- Toy setup similar to that of **Theisen, Klusowski, Mahoney**

$$
\begin{aligned}
(\mathbf{x}_i, y_i) \sim_{\text{iid}} \frac{1}{2}(N_+, 1) + \frac{1}{2}(N_-, -1) \quad i = 1, \ldots, n \\
N_+ = \mathcal{N}(\mu, I_d) \\
N_- = \mathcal{N}(-\mu, I_d)
\end{aligned}
$$

- [Muthukumar *et al*, 2020] and [H., Muthukumar, Xu]: If $d \gg n \log n$, then with high probability, every training example is a support vector:

$$\mathbf{x}_i^\top w_{\text{svm}} = y_i, \quad i = 1, \ldots, n$$

where w_{svm} is the SVM solution.

Margins in very high-dimensions

- Toy setup similar to that of **Theisen, Klusowski, Mahoney**

$$
\begin{aligned}
(\mathbf{x}_i, y_i) \sim_{\text{iid}} \frac{1}{2}(N_+, 1) + \frac{1}{2}(N_-, -1) \quad i = 1, \dots, n \\
N_+ &= \mathcal{N}(\mu, I_d) \\
N_- &= \mathcal{N}(-\mu, I_d)
\end{aligned}
$$

- [Muthukumar *et al*, 2020] and [H., Muthukumar, Xu]:
  If $d \gg n \log n$, then with high probability, every training example is a support vector:

$$\mathbf{x}_i^\mathsf{T} w_{\text{svm}} = y_i, \quad i = 1, \dots, n$$

  where w_{svm} is the SVM solution.
  - In this case: *Minimum norm interpolation* $=$ *SVM solution*.

Margins in very high-dimensions

- Toy setup similar to that of **Theisen, Klusowski, Mahoney**

$$(\mathbf{x}_i, y_i) \sim_{\text{iid}} \frac{1}{2}(N_+, 1) + \frac{1}{2}(N_-, -1) \quad i = 1, \ldots, n$$
$$N_+ = \mathcal{N}(\mu, I_d)$$
$$N_- = \mathcal{N}(-\mu, I_d)$$

- [Muthukumar *et al*, 2020] and [H., Muthukumar, Xu]:
  If $d \gg n \log n$, then with high probability, every training example is a support vector:

$$\mathbf{x}_i^\mathsf{T} w_{\text{svm}} = y_i, \quad i = 1, \ldots, n$$

  where w_{svm} is the SVM solution.
  - In this case: *Minimum norm interpolation = SVM solution*.
  - (Similar behavior under anisotropic (subgaussian or subsampled Haar) designs if covariance eigenvalues decay slowly enough)

Margins in very high-dimensions

- Toy setup similar to that of **Theisen, Klusowski, Mahoney**

$$
\begin{aligned}
(\mathbf{x}_i, y_i) \sim_{\text{iid}} \frac{1}{2}(N_+, 1) + \frac{1}{2}(N_-, -1) \quad i = 1, \dots, n \\
N_+ = \mathcal{N}(\mu, I_d) \\
N_- = \mathcal{N}(-\mu, I_d)
\end{aligned}
$$

- [Muthukumar *et al*, 2020] and [H., Muthukumar, Xu]:
  If $d \gg n \log n$, then with high probability, every training example is a support vector:

$$
\mathbf{x}_i^{\mathsf{T}} w_{\text{svm}} = y_i, \quad i = 1, \dots, n
$$

  where w_{svm} is the SVM solution.
  - In this case: *Minimum norm interpolation = SVM solution*.
  - (Similar behavior under anisotropic (subgaussian or subsampled Haar) designs if covariance eigenvalues decay slowly enough)
- What about kernels that matter?

ReLU nets in the Neural Tangent Kernel (NTK) regime

Goal: Prove error rate $\epsilon \in (0, 1)$ is achieved by training a neural net via gradient descent on an empirical risk

ReLU nets in the Neural Tangent Kernel (NTK) regime

Goal: Prove error rate $\epsilon \in (0, 1)$ is achieved by training a neural net via gradient descent on an empirical risk

- NTK: kernel-based characterization of certain training processes

ReLU nets in the Neural Tangent Kernel (NTK) regime

Goal: Prove error rate $\epsilon \in (0, 1)$ is achieved by training a neural net via gradient descent on an empirical risk

- NTK: kernel-based characterization of certain training processes
- Flurry of results about wide ReLU nets in the NTK regime

ReLU nets in the Neural Tangent Kernel (NTK) regime

Goal: Prove error rate $\epsilon \in (0, 1)$ is achieved by training a neural net via gradient descent on an empirical risk

- NTK: kernel-based characterization of certain training processes
- Flurry of results about wide ReLU nets in the NTK regime
  - $\mathrm{poly}(1/\epsilon)$ width is sufficient for *regression* on noise-free data

ReLU nets in the Neural Tangent Kernel (NTK) regime

Goal: Prove error rate $\epsilon \in (0, 1)$ is achieved by training a neural net via gradient descent on an empirical risk

- NTK: kernel-based characterization of certain training processes
- Flurry of results about wide ReLU nets in the NTK regime
  - $\mathrm{poly}(1/\epsilon)$ width is sufficient for *regression* on noise-free data
- **Ji and Telgarsky**:

ReLU nets in the Neural Tangent Kernel (NTK) regime

Goal: Prove error rate $\epsilon \in (0, 1)$ is achieved by training a neural net via gradient descent on an empirical risk

- NTK: kernel-based characterization of certain training processes
- Flurry of results about wide ReLU nets in the NTK regime
  - $\mathrm{poly}(1/\epsilon)$ width is sufficient for *regression* on noise-free data
- **Ji and Telgarsky**:
  - $\mathrm{poly}\log(1/\epsilon)$ width is sufficient for *classification* on data separable with a margin

ReLU nets in the Neural Tangent Kernel (NTK) regime

Goal: Prove error rate $\epsilon \in (0, 1)$ is achieved by training a neural net via gradient descent on an empirical risk

- NTK: kernel-based characterization of certain training processes
- Flurry of results about wide ReLU nets in the NTK regime
  - $\mathrm{poly}(1/\epsilon)$ width is sufficient for *regression* on noise-free data
- **Ji and Telgarsky**:
  - $\mathrm{poly}\log(1/\epsilon)$ width is sufficient for *classification* on data separable with a margin
  - Margin is defined with respect to infinite-width NTK

ReLU nets in the Neural Tangent Kernel (NTK) regime

Goal: Prove error rate $\epsilon \in (0, 1)$ is achieved by training a neural net via gradient descent on an empirical risk

- NTK: kernel-based characterization of certain training processes
- Flurry of results about wide ReLU nets in the NTK regime
  - $\mathrm{poly}(1/\epsilon)$ width is sufficient for *regression* on noise-free data
- **Ji and Telgarsky**:
  - $\mathrm{poly}\log(1/\epsilon)$ width is sufficient for *classification* on data separable with a margin
  - Margin is defined with respect to infinite-width NTK
- A pressing question: Is gap the real?

ReLU nets in the Neural Tangent Kernel (NTK) regime

Goal: Prove error rate $\epsilon \in (0, 1)$ is achieved by training a neural net via gradient descent on an empirical risk

- NTK: kernel-based characterization of certain training processes
- Flurry of results about wide ReLU nets in the NTK regime
  - $\text{poly}(1/\epsilon)$ width is sufficient for *regression* on noise-free data
- **Ji and Telgarsky**:
  - $\text{poly}\log(1/\epsilon)$ width is sufficient for *classification* on data separable with a margin
  - Margin is defined with respect to infinite-width NTK
- A pressing question: Is gap the real?
  - (Telgarsky: "It's subtle ...")

Parting words

Uniform convergence $+$ ERM go hand-in-hand

Parting words

Uniform convergence + ERM go hand-in-hand

- Uniform convergence makes sense $\Leftrightarrow$ empirical risk makes sense
- If empirical risk $= 0$ always, maybe look elsewhere

Parting words

Uniform convergence + ERM go hand-in-hand

- Uniform convergence makes sense $\Leftrightarrow$ empirical risk makes sense
- If empirical risk $= 0$ always, maybe look elsewhere

Shift focus of analysis to inductive bias (e.g., margins)

Parting words

Uniform convergence + ERM go hand-in-hand

- Uniform convergence makes sense $\Leftrightarrow$ empirical risk makes sense
- If empirical risk $= 0$ always, maybe look elsewhere

Shift focus of analysis to inductive bias (e.g., margins)

Thank you!

Thanks to: NSF CCF-1740833, DMR-153491; Sloan Research Fellowship; Simons Institute for the Theory of Computing Sp'17 & Su'19 programs