Nonasymptotic Theory for Two-Layer Neural Networks: Beyond the Bias–Variance Trade-Off

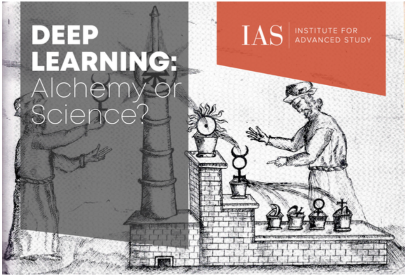
Wei Lin
Peking University

Joint work with Huiyuan Wang

BIRS-IASM Workshop, Hangzhou
December 14, 2023

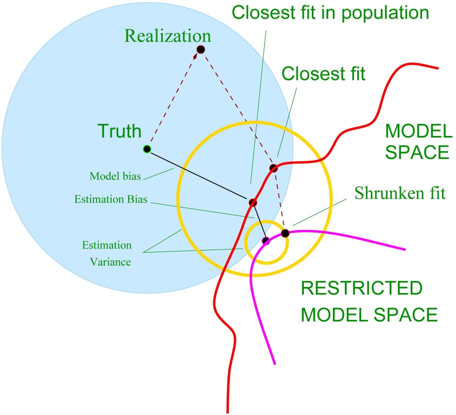
Deep Learning: Alchemy or Science?

“Deep learning has led to dramatic progress on problems of artificial intelligence . . . and triggered a new gold rush in the tech sector. Some researchers have raised the concern that the rapid progress has led to loss of rigor and precision.”



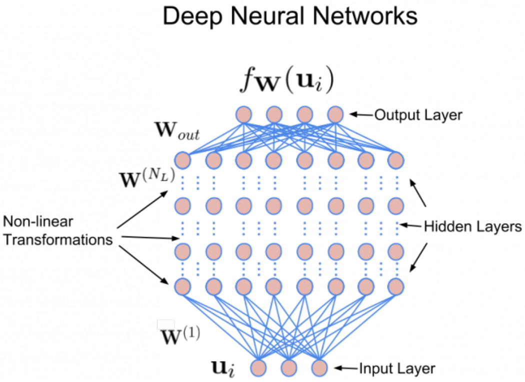
Toward Deep Science: Generalization Guarantees

- Generalization/prediction/out-of-sample/test error: measure of how accurately an algorithm predicts an outcome on previously unseen data
- Decomposition of generalization error
  - Approximation error (bias)
  - Estimation error (variance)
  - Optimization error



Deep Neural Networks

- Important features
  - Compositional structure
  - Activation function (e.g., ReLU)
  - Deeper vs. wider
- Comparison with classical statistical/ML models
  - Linear models
  - Fully nonparametric models
  - Additive models
  - Single-index models
  - Multi-index models



Two-Layer ReLU Networks

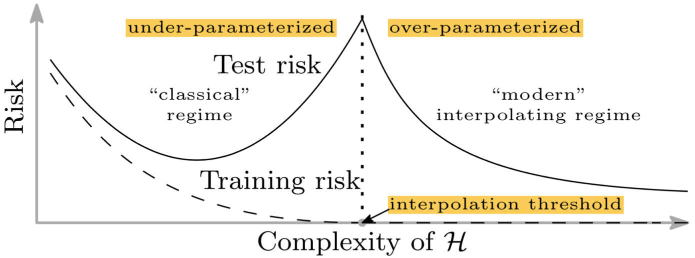
- Consider a two-layer ReLU network $g(\cdot;\boldsymbol{\theta}) : \mathbb{R}^d \to \mathbb{R}$ with m hidden units:
$$g(\boldsymbol{x};\boldsymbol{\theta}) = \sum_{k=1}^{m} a_k \sigma(\boldsymbol{v}_k^T \boldsymbol{x} + b_k),$$
where the parameter $\boldsymbol{\theta} = (a_1, \ldots, a_m, \boldsymbol{v}_1^T, \ldots, \boldsymbol{v}_m^T, b_1, \ldots, b_m)^T$ and $\sigma(z) = \max(z, 0)$
- Why is the theory nontrivial?
  - Nonidentifiability: consistency in parameter estimation is impossible
  - Nonconvexity: global or local, which optimum?
  - Overparametrization: no complexity control via sparsity

Related Work

- Limiting behavior of two-layer networks as $m \to \infty$
  - Mean field approximation (Mei, Montanari and Nguyen, *PNAS*, 2018)
  - Neural tangent kernel (Jacot, Gabriel and Hongler, *NeurIPS*, 2018)
- L_2 risk bounds for two-layer networks with explicit regularization
  - Barron (*MLJ*, 1994): $\dfrac{1}{m} + \dfrac{md \log n}{n}$ Classical bias–variance trade-off
  - E, Ma and Wu (*Comm. Math. Sci.*, 2019): $\dfrac{1}{m} + \log n \sqrt{\dfrac{\log d}{n}}$ No trade-off!
  - Parhi and Nowak (*TIT*, 2023): $n^{-(d+3)/(2d+3)}$ Minimax optimal, but underparametrized
- Nonasymptotic bounds for deep neural networks
  - Schmidt-Hieber (*AOS*, 2020): compositional function class
  - Farrell, Liang and Misra (*Econometrica*, 2021): Hölder class

Overparametrization

- Classical bias–variance trade-off achieved by complexity control
- The double descent phenomenon (Belkin et al., *PNAS*, 2019)



Q1. How does the network perform in the overparametrized regime differently from in the underparametrized regime?

Q2. How does the overparametrized minimum risk compare with its underparametrized counterpart and how far is it from optimal?

This Work

- A generalization theory for two-layer ReLU networks
  - Explicit regularization: no sparsity
  - Algorithm-independent: for any global optimum
  - Nonasymptotic bounds: for any finite n and m
  - Minimax lower bounds: achieved in the infinite-width limit
  - Random feature models: curse of dimensionality, suboptimal

Target Function Class

- The functions of interest lie in the space

$$\mathcal{G} = \Big\{ f\colon \boldsymbol{x} \mapsto \int_{\mathbb{R}^{d+1}} \big(\sigma(\boldsymbol{v}^T\boldsymbol{x} + b) - \sigma(b)\big)\, d\alpha(\boldsymbol{w}) : \|f\|_{\mathcal{S}} \equiv \int_{\mathbb{R}^{d+1}} \|\boldsymbol{v}\|_2 \, d|\alpha|(\boldsymbol{w}) < \infty \Big\},$$

  where $\boldsymbol{w} = (\boldsymbol{v}^T, b)^T$ and α is a signed measure
- Approximation limits for finitely wide ReLU networks

$$g(\boldsymbol{x};\boldsymbol{\theta}) = \int_{\mathbb{R}^{d+1}} \big(\sigma(\boldsymbol{v}^T\boldsymbol{x} + b) - \sigma(b)\big)\, d\alpha_m(\boldsymbol{w}) + g(\boldsymbol{0};\boldsymbol{\theta}),$$

  where $\alpha_m = \sum_{k=1}^{m} a_k \delta_{\boldsymbol{w}_k}$

Model and Assumptions

- Consider the data-generating model

$$y_i = f^*(\boldsymbol{x}_i) + \varepsilon_i, \quad i = 1, \ldots, n$$

- Assumptions
  1. $f^* \in \mathcal{G}_M \equiv \{f \in \mathcal{G} : \|f\|_{\mathcal{S}} \leq M\}$ for some constant $M > 0$
  2. $\boldsymbol{x}_i \sim \mu$ independently, where μ is supported in $\mathbb{B}^d$
  3. $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$ independently and are independent of $\boldsymbol{x}_i$

Scaled Variation Regularization

- For any finitely wide two-layer ReLU network $g(\cdot;\boldsymbol{\theta})$, define the scaled variation regularizer
$$\nu(\boldsymbol{\theta}) = \sum_{k=1}^{m} |a_k| \|\boldsymbol{w}_k\|_2$$
where $\boldsymbol{w}_k = (\boldsymbol{v}_k^T, b_k)^T$
- Consider the regularized empirical risk minimization problem
$$\widehat{\boldsymbol{\theta}} = \operatorname*{arg\,min}_{\boldsymbol{\theta} \in \boldsymbol{\Theta}_m} \left\{ \frac{1}{2n} \sum_{i=1}^{n} \big(y_i - g(\boldsymbol{x}_i;\boldsymbol{\theta})\big)^2 + \lambda \nu(\boldsymbol{\theta}) \right\}$$

Approximation Rates

- The approximation rate of $O(m^{-1/2})$ in Barron (*TIT*, 1993) can be improved by using Bach (*JMLR*, 2017):

  Theorem. For any $f \in \mathcal{G}_M$, there exists a network $g(\cdot; \boldsymbol{\theta})$ of width m such that $\nu(\boldsymbol{\theta}) \leq 6\|f\|_{\mathcal{S}}$ and

  $$\|f - g(\cdot; \boldsymbol{\theta})\|_{L_\infty(\mathbb{B}^d)} \leq C\|f\|_{\mathcal{S}} m^{-(d+3)/(2d)}$$

  for some constant $C > 0$ depending only on d.

Equivalence to Ridge Regression

- By the reparametrization $\tilde{\boldsymbol{\theta}} = \mathcal{T}_1(\boldsymbol{\theta})$

$$\tilde{a}_k = a_k \sqrt{\frac{\|\boldsymbol{w}_k\|_2}{|a_k|}}, \qquad \tilde{\boldsymbol{w}}_k = \boldsymbol{w}_k \sqrt{\frac{|a_k|}{\|\boldsymbol{w}_k\|_2}},$$

the scaled variation regularizer $\nu(\cdot)$ becomes the ℓ_2/ridge/weight decay penalty

$$\frac{1}{2} \sum_{k=1}^{m} (\tilde{a}_k^2 + \|\tilde{\boldsymbol{w}}_k\|_2^2)$$

Equivalence to Ridge Regression

- Proposition. Any solution $\widehat{\boldsymbol{\theta}}_{\ell_2}$ to the ℓ_2-regularized problem is a solution to the network estimation problem. Conversely, if $\widehat{\boldsymbol{\theta}}$ is a solution to the network estimation problem, then $\mathcal{T}_1(\widehat{\boldsymbol{\theta}})$ is a solution to the ℓ_2-regularized problem.

- Proposition. Consider the gradient flows

$$\frac{d}{dt}\boldsymbol{\theta}(t) = -\nabla_{\boldsymbol{\theta}} J_n(\boldsymbol{\theta}(t); \lambda)$$

for the two problems, both initialized at $\boldsymbol{\theta}(0) = \mathcal{T}_1(\boldsymbol{\theta}_0)$ for an arbitrary $\boldsymbol{\theta}_0 \in \boldsymbol{\Theta}_m$. Then the trajectories of the two gradient flows coincide.

Connection to Group Lasso

- Important observation: The n hyperplanes $\boldsymbol{x}_i^T \boldsymbol{v} + b = 0$ divide the parameter space $\mathbb{R}^{n+1}$ into finitely many regions $R_1, \ldots, R_p$, so that $\boldsymbol{D} = \mathrm{diag}(I(\boldsymbol{X}\boldsymbol{w} \geq 0))$ stays constant over each R_j; the number of these regions

$$p \leq 2 \sum_{j=0}^{d} \binom{n-1}{j} \leq 2n^d,$$

  where the first upper bound is sharp when $\boldsymbol{X}$ has full rank

- Taking into account the sign of a, we partition the parameter space $\mathbb{R}^{d+2}$ into $2p$ regions

$$Q_j = [0, \infty) \times R_j, \quad Q_{p+j} = (-\infty, 0) \times R_j, \quad j = 1, \ldots, p$$

Connection to Group Lasso

- The linearity of ReLU over each R_j and optimality of $\widehat{\boldsymbol{\theta}}$ entail:

  Proposition. For any network estimator $\widehat{\boldsymbol{\theta}}$, if $(\widehat{a}_k, \widehat{\boldsymbol{w}}_k^T)^T$ and $(\widehat{a}_\ell, \widehat{\boldsymbol{w}}_\ell^T)^T$ lie in the same cone Q_j, then $\widehat{\boldsymbol{w}}_k$ and $\widehat{\boldsymbol{w}}_\ell$ must be collinear, that is, $\widehat{\boldsymbol{w}}_k = c_0 \widehat{\boldsymbol{w}}_\ell$ for some constant $c_0 > 0$.

Connection to Group Lasso

- We therefore collect the weights in the same cone and reformulate the problem into a group lasso:

  Proposition. The network estimator $\widehat{\boldsymbol{\theta}}$ satisfies

$$J_n(\widehat{\boldsymbol{\theta}};\lambda) = \frac{1}{2n}\left\| \boldsymbol{y} - \sum_{j=1}^{2p} \boldsymbol{D}_j \boldsymbol{X} \boldsymbol{\beta}_j(\widehat{\boldsymbol{\theta}}) \right\|_2^2 + \lambda \|\boldsymbol{B}(\widehat{\boldsymbol{\theta}})\|_{2,1},$$

  where

$$\boldsymbol{\beta}_j(\boldsymbol{\theta}) = \sum_{k:(a_k,\boldsymbol{w}_k^T)^T \in Q_j} |a_k| \boldsymbol{w}_k, \quad \|\boldsymbol{B}(\boldsymbol{\theta})\|_{2,1} = \sum_{j=1}^{2p} \|\boldsymbol{\beta}_j(\boldsymbol{\theta})\|_2.$$

Generalization Bounds

- Theorem. Under Conditions 1–3, the network estimator $g(\cdot;\widehat{\boldsymbol{\theta}})$ with $\lambda = C_1\sigma_\varepsilon \min\{\sqrt{d\log n/n}, \max(m^{-(d+3)/d}, md\log n/n)\}$ satisfies

$$\|g(\cdot;\widehat{\boldsymbol{\theta}}) - f^*\|_2^2 \le C\Bigg\{\|f^*\|_{\mathcal{S}}^2 m^{-(d+3)/d} + (\sigma_\varepsilon^2 + \|f^*\|_{\mathcal{S}}^2)\min\left(\sqrt{\frac{d\log n}{n}}, \frac{md\log n}{n}\right)\Bigg\}$$

with probability at least $1 - O(n^{-C_2})$ for some constants $C_1, C_2, C > 0$.

Generalization Bounds

- First or underparametrized valley occurs at $m_0 \asymp (n/(d \log n))^{d/(2d+3)}$ with minimum risk $O((d \log n/n)^{(d+3)/(2d+3)})$
- Second or overparametrized valley occurs at $m \to \infty$ with minimum risk $O(\sqrt{d \log n/n})$
- Critical point $m_1 \asymp \sqrt{n/(d \log n)}$, after which the model complexity and hence the variance remain constant

Double Descent Curve

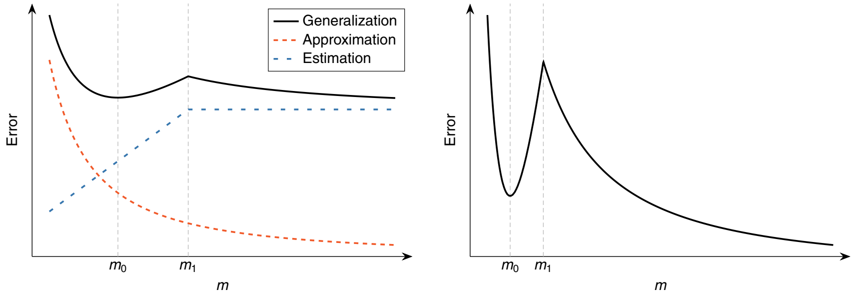


Figure: Risk curves for varying network width m with $\|f^*\|_{\mathcal{S}}^2/\sigma_\varepsilon^2 = 1$, $d = 6$, and $n = 1000$

Double Descent Curve

- Asymptotically, the underparametrized valley is lower:

$$O\left(\left(\frac{d\log n}{n}\right)^{(d+3)/(2d+3)}\right) \quad \text{vs.} \quad O\left(\sqrt{\frac{d\log n}{n}}\right),$$

  with the gap vanishing as $d \to \infty$
- In finite samples, the overparametrized valley is lower whenever

$$\kappa \equiv \frac{\|f^*\|_{\mathcal{S}}^2}{\sigma_\varepsilon^2 + \|f^*\|_{\mathcal{S}}^2} > \left(\frac{1}{2}\right)^{(2d+3)/d} \left(\frac{n}{d\log n}\right)^{3/(2d)}$$

- When $d \gg \log n$, this approximately requires $\kappa > 1/4$, or the signal-to-noise ratio $\|f^*\|_{\mathcal{S}}^2/\sigma_\varepsilon^2 = \kappa/(1-\kappa) > 1/3$

Minimax Lower Bounds

- The underparametrized minimum risk has been shown to be minimax optimal over $\mathcal{G}_M$ (Parhi and Nowak, *TIT*, 2023)
- The overparametrized minimum risk, however, is also minimax optimal, over a slightly larger class of functions:

  Theorem. Assume that $\boldsymbol{x}_i \sim \text{Uniform}(\mathbb{B}^d)$ and $\varepsilon_i \sim N(0,1)$. Then there exists a constant $C > 0$ such that

$$\inf_{\widehat{f}} \sup_{f^* \in \mathcal{G}} \mathbb{E}\|\widehat{f} - f^*\|_2^2 \geq \frac{C}{\sqrt{n \log n}},$$

  where the infimum is taken over all estimators.

Random Feature Models

- Random feature models provide a stochastic approximation to kernel methods:

$$h_{\rho_0}(\boldsymbol{x}; \boldsymbol{a}) = \frac{1}{\sqrt{m}} \sum_{k=1}^{m} a_k \sigma(\boldsymbol{v}_k^T \boldsymbol{x} + b_k),$$

  where $\boldsymbol{w}_k = (\boldsymbol{v}_k^T, b_k)^T \sim \rho_0$ independently for some fixed ρ_0 and only $\boldsymbol{a} = (a_1, \ldots, a_m)^T$ needs to be estimated
- Mei and Montanari (*CPAM*, 2022) showed the double descent curve for random feature models when $m, n, d \to \infty$ with $m \asymp n \asymp d$

Random Feature Models

- However, random feature models suffer from the curse of dimensionality and is suboptimal over $\mathcal{G}_M$:

  Proposition. Under Conditions 1 and 3, there exists a universal constant $C > 0$ such that

$$\sup_{f^* \in \mathcal{G}_M} \mathbb{E}\| h_{\rho_0}(\cdot; \widehat{\boldsymbol{a}}) - f^* \|_2^2 \geq \frac{CM}{d\{\min(m, n)\}^{1/d}}.$$

Discussion

- Unique insights from our results
  - Impact of dimensionality
  - Double descent with optimal regularization
  - Complexity control
  - Bias–variance trade-off (Derumigny and Schmidt-Hieber, *AOS*, 2023)
- Future work
  - Deep neural networks
  - Implicit regularization: noise injection, early stopping, etc.
  - Classification problems
  - More architectures: CNN, RNN, ResNet, etc.

Available at arXiv:2106.04795v2

Thank You!