From kernel methods to neural networks: double descent, function spaces, curse of dimensionality

Fanghui Liu

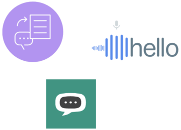
Department of Computer Science, University of Warwick, UK
Centre for Discrete Mathematics and its Applications (DIMAP), Warwick

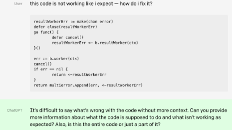
Based on joint work with

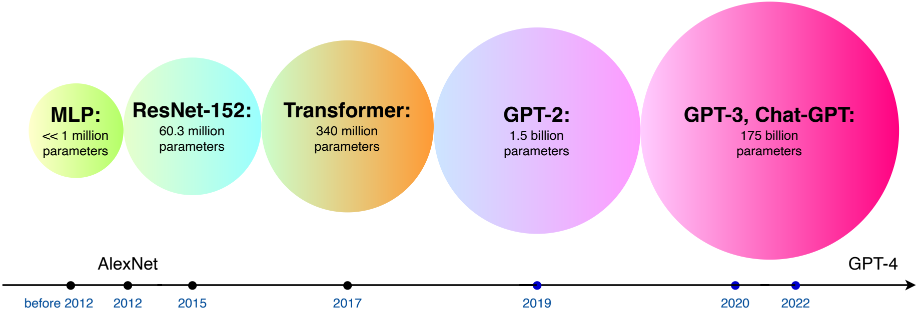
[Johan A.K. Suykens (KU Leuven), Volkan Cevher (EPFL)]

at Department of Statistics, University of Warwick

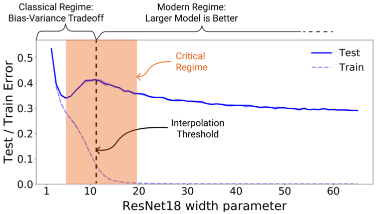
Over-parameterization: more parameters than training data



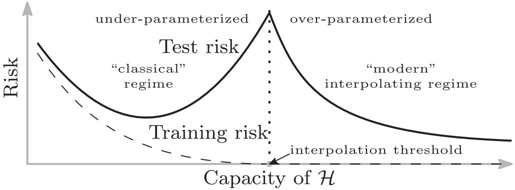




Surprises in modern neural networks: double descent

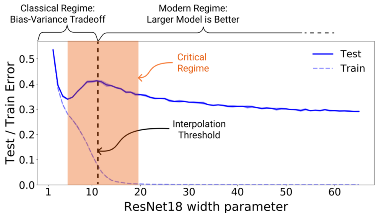


(a) Training and test error on ResNet18 [1]

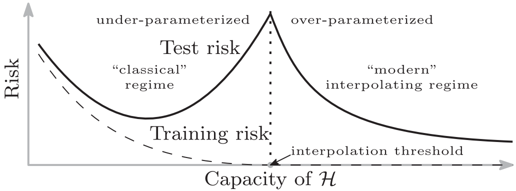


(b) Double descent [2] (Belkin, Hsu, Ma, Mandal, 2019).

Surprises in modern neural networks: double descent



(a) Training and test error on ResNet18 [1]

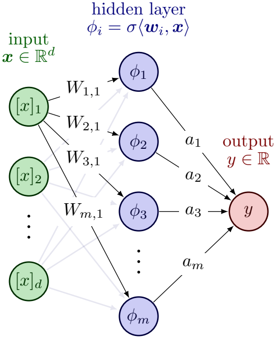


(b) Double descent [2] (Belkin, Hsu, Ma, Mandal, 2019).

Observations: beyond bias-variance trade-off

- 1) Monotonic decreasing in the overparameterized regime
- 2) Global minimum when #parameters is infinite
- 3) Peak at the interpolation thresholds

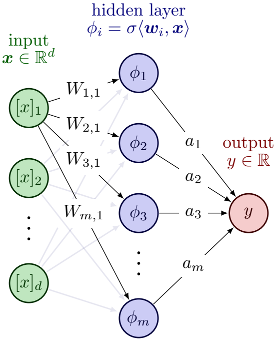
Background: Two-layer neural networks



$$f_m(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{i=1}^{m} a_i \phi(\boldsymbol{x}, \boldsymbol{w}_i), \quad \boldsymbol{\theta} := \{(a_i, \boldsymbol{w}_i)\}_{i=1}^{m}$$

- $\phi : \mathcal{X} \times \mathcal{W} \to \mathbb{R}$, e.g., ReLU: $\phi(\boldsymbol{x}, \boldsymbol{w}) = \max(\langle \boldsymbol{x}, \boldsymbol{w} \rangle, 0)$

Background: Two-layer neural networks



$$f_m(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{i=1}^{m} a_i \phi(\boldsymbol{x}, \boldsymbol{w}_i), \quad \boldsymbol{\theta} := \{(a_i, \boldsymbol{w}_i)\}_{i=1}^m$$

- $\phi : \mathcal{X} \times \mathcal{W} \to \mathbb{R}$, e.g., ReLU: $\phi(\boldsymbol{x}, \boldsymbol{w}) = \max(\langle \boldsymbol{x}, \boldsymbol{w} \rangle, 0)$
- Random features models (RFMs) [3]:
  - $\{\boldsymbol{w}_i\}_{i=1}^m \overset{iid}{\sim} \mu$ for a given $\mu \in \mathcal{P}(\mathcal{W})$
  - only train the second layer

Recall RFMs in high-dimensional asymptotic setting (Mei and Montanari, 2022)

∘ random feature regression with $\widehat{\boldsymbol{a}}_\lambda = \arg\min_{\boldsymbol{a}} \widehat{\mathcal{E}}_\lambda(\boldsymbol{a})$

$$\widehat{\mathcal{E}}_\lambda(\boldsymbol{a}) = \frac{1}{n}\sum_{i=1}^{n}\left[y_i - \frac{1}{m}\sum_{j=1}^{m} a_j \sigma(\langle \boldsymbol{x}_i, \boldsymbol{w}_j \rangle)\right]^2 + \frac{\lambda m}{d}\|\boldsymbol{a}\|_2^2$$

Recall RFMs in high-dimensional asymptotic setting (Mei and Montanari, 2022)

- random feature regression with $\widehat{\boldsymbol{a}}_\lambda = \arg\min_{\boldsymbol{a}} \widehat{\mathcal{E}}_\lambda(\boldsymbol{a})$

$$\widehat{\mathcal{E}}_\lambda(\boldsymbol{a}) = \frac{1}{n} \sum_{i=1}^{n} \left[y_i - \frac{1}{m} \sum_{j=1}^{m} a_j \sigma(\langle \boldsymbol{x}_i, \boldsymbol{w}_j \rangle) \right]^2 + \frac{\lambda m}{d} \|\boldsymbol{a}\|_2^2$$

$$\mathcal{E}(\boldsymbol{a}, f_\rho) = \mathbb{E}_{\boldsymbol{x},y} \left[f_\rho(\boldsymbol{x}) - \frac{1}{m} \sum_{j=1}^{m} a_j \sigma(\langle \boldsymbol{x}_i, \boldsymbol{w}_j \rangle) \right]^2$$

Recall RFMs in high-dimensional asymptotic setting (Mei and Montanari, 2022)

∘ random feature regression with $\widehat{\boldsymbol{a}}_\lambda = \arg\min_{\boldsymbol{a}} \widehat{\mathcal{E}}_\lambda(\boldsymbol{a})$

$$\widehat{\mathcal{E}}_\lambda(\boldsymbol{a}) = \frac{1}{n}\sum_{i=1}^{n}\left[y_i - \frac{1}{m}\sum_{j=1}^{m} a_j \sigma(\langle \boldsymbol{x}_i, \boldsymbol{w}_j \rangle)\right]^2 + \frac{\lambda m}{d}\|\boldsymbol{a}\|_2^2$$

$$\mathcal{E}(\boldsymbol{a}, f_\rho) = \mathbb{E}_{\boldsymbol{x},y}\left[f_\rho(\boldsymbol{x}) - \frac{1}{m}\sum_{j=1}^{m} a_j \sigma(\langle \boldsymbol{x}_i, \boldsymbol{w}_j \rangle)\right]^2$$

Theorem (double descent of RFMs [4])

Under proper assumptions, if target function is linear, under the high-dimensional setting

- $n, m, d \to \infty$, $m/d \to \psi_1$ *and* $n/d \to \psi_2$ *as* $d \to \infty$ *with* $\psi_1, \psi_2 \in (0, \infty)$

Recall RFMs in high-dimensional asymptotic setting (Mei and Montanari, 2022)

◦ random feature regression with $\widehat{\boldsymbol{a}}_\lambda = \arg\min_{\boldsymbol{a}} \widehat{\mathcal{E}}_\lambda(\boldsymbol{a})$

$$\widehat{\mathcal{E}}_\lambda(\boldsymbol{a}) = \frac{1}{n}\sum_{i=1}^{n}\left[y_i - \frac{1}{m}\sum_{j=1}^{m} a_j \sigma(\langle \boldsymbol{x}_i, \boldsymbol{w}_j \rangle)\right]^2 + \frac{\lambda m}{d}\|\boldsymbol{a}\|_2^2$$

$$\mathcal{E}(\boldsymbol{a}, f_\rho) = \mathbb{E}_{\boldsymbol{x},y}\left[f_\rho(\boldsymbol{x}) - \frac{1}{m}\sum_{j=1}^{m} a_j \sigma(\langle \boldsymbol{x}_i, \boldsymbol{w}_j \rangle)\right]^2$$

Theorem (double descent of RFMs [4])

Under proper assumptions, if target function is linear, under the high-dimensional setting

- $n, m, d \to \infty$, $m/d \to \psi_1$ *and* $n/d \to \psi_2$ *as* $d \to \infty$ *with* $\psi_1, \psi_2 \in (0, \infty)$

$$\mathcal{E}(\widehat{\boldsymbol{a}}_\lambda, f_\rho) = \textit{Bias} + \textit{Variance} + o_{d,\mathbb{P}}(1)\,.$$

observations 1), 2), 3) for double descent can be theoretically proved.

Questions on high dimensional kernel methods

high dimensional kernel methods: can only learn linear function! [5] (Ghorbani, Mei, Misiakiewicz, Montanari, 2021)

Questions on high dimensional kernel methods

high dimensional kernel methods: can only learn linear function! [5] (Ghorbani, Mei, Misiakiewicz, Montanari, 2021)

- asymptotic expansion under high dimensions [6] (El Karoui, 2010)
  under the setting of $n, d \to \infty$, $n/d \to \psi_1$ as $d \to \infty$ with $\psi_1 \in (0, \infty)$, we have

$$\|\boldsymbol{K} - (a\boldsymbol{X}\boldsymbol{X}^\top + b\boldsymbol{I})\|_2 \xrightarrow{\mathbb{P}} 0 \quad \text{when } d \to \infty \quad \text{for some parameters } a, b$$

Questions on high dimensional kernel methods

high dimensional kernel methods: can only learn linear function! [5] (Ghorbani, Mei, Misiakiewicz, Montanari, 2021)

- asymptotic expansion under high dimensions [6] (El Karoui, 2010)
  under the setting of $n, d \to \infty$, $n/d \to \psi_1$ as $d \to \infty$ with $\psi_1 \in (0, \infty)$, we have

$$\|\boldsymbol{K} - (a\boldsymbol{X}\boldsymbol{X}^\top + b\boldsymbol{I})\|_2 \xrightarrow{\mathbb{P}} 0 \text{ when } d \to \infty \quad \text{for some parameters } a, b$$

- $\|f\|_{\mathcal{H}} < \infty$?

Example (a linear function $f : \mathbb{S}^d \to \mathbb{R}$ such that $f(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x}$ for a certain $\boldsymbol{v} \in \mathbb{S}^d$)

- zero-order arc-cosine kernel $k(\boldsymbol{x}, \boldsymbol{x}') = \int_{\mathbb{S}^d} 1_{\{\boldsymbol{w}^\top \boldsymbol{x} \geq 0\}} 1_{\{\boldsymbol{w}^\top \boldsymbol{x}' \geq 0\}} \mathrm{d}\mu(\boldsymbol{w})$
  $\Rightarrow \|f\|_{\mathcal{H}} = \frac{2d\pi}{d-1}\pi < 4\pi$ [7] (Bach 2017)

Questions on high dimensional kernel methods

high dimensional kernel methods: can only learn linear function! [5] (Ghorbani, Mei, Misiakiewicz, Montanari, 2021)

- asymptotic expansion under high dimensions [6] (El Karoui, 2010)
  under the setting of $n, d \to \infty$, $n/d \to \psi_1$ as $d \to \infty$ with $\psi_1 \in (0, \infty)$, we have

$$\|\boldsymbol{K} - (a\boldsymbol{X}\boldsymbol{X}^\top + b\boldsymbol{I})\|_2 \xrightarrow{\mathbb{P}} 0 \quad \text{when } d \to \infty \quad \text{for some parameters } a, b$$

- $\|f\|_{\mathcal{H}} < \infty$?

Example (a linear function $f : \mathbb{S}^d \to \mathbb{R}$ such that $f(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x}$ for a certain $\boldsymbol{v} \in \mathbb{S}^d$)

- zero-order arc-cosine kernel $k(\boldsymbol{x}, \boldsymbol{x}') = \int_{\mathbb{S}^d} 1_{\{\boldsymbol{w}^\top \boldsymbol{x} \geq 0\}} 1_{\{\boldsymbol{w}^\top \boldsymbol{x}' \geq 0\}} \mathrm{d}\mu(\boldsymbol{w})$
  $\Rightarrow \|f\|_{\mathcal{H}} = \frac{2d\pi}{d-1}\pi < 4\pi$ [7] (Bach 2017)
- first-order arc-cosine kernel, we have $\|f\|_{\mathcal{H}} \asymp C\sqrt{d}$ for some constant C independent of d.

Motivation

- high dimension vs. fixed dimension
- from asymptotic to non-asymptotic
- two-layer neural networks trained by SGD

Motivation

- high dimension vs. fixed dimension
- from asymptotic to non-asymptotic
- two-layer neural networks trained by SGD

○ Analysis

- SGD: implicit regularization $\rightarrow$ without λ
- dimension-free SGD bound
- multiple randomness sources
  - data sampling, label noise, Gaussian initialization, stochastic gradients

Motivation

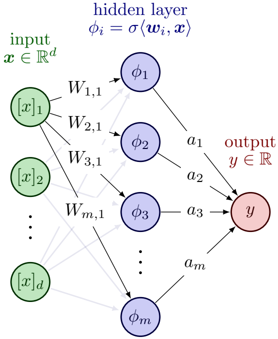
- high dimension vs. fixed dimension
- from asymptotic to non-asymptotic
- two-layer neural networks trained by SGD

◦ Analysis

- SGD: implicit regularization $\to$ without λ
- dimension-free SGD bound
- multiple randomness sources
  - data sampling, label noise, Gaussian initialization, stochastic gradients

observations 1), 2), 3) can be still proved!

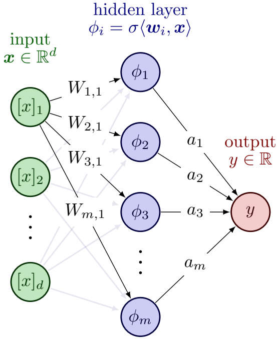
Problem settings: function space



random features mapping:

$$\varphi(\boldsymbol{x}) := \frac{1}{\sqrt{m}} \sigma\left(\frac{\boldsymbol{W}\boldsymbol{x}}{\sqrt{d}}\right) \quad W_{ij} \sim \mathcal{N}(0,1)$$

Problem settings: function space



random features mapping:

$$\varphi(\boldsymbol{x}) := \frac{1}{\sqrt{m}}\sigma\left(\frac{\boldsymbol{W}\boldsymbol{x}}{\sqrt{d}}\right) \quad W_{ij} \sim \mathcal{N}(0,1)$$

function space

$$\mathcal{H} := \left\{ f \in L^2_{\rho_X} \,\middle|\, f(\boldsymbol{x}) = \langle \boldsymbol{a}, \varphi(\boldsymbol{x})\rangle \right\}, \quad \boldsymbol{W}_{ij} \sim \mathcal{N}(0,1)$$

covariance operator: $\Sigma_m := \mathbb{E}_{\boldsymbol{x}}[\varphi(\boldsymbol{x}) \otimes \varphi(\boldsymbol{x})]$

expected covariance operator: $\widetilde{\Sigma}_m := \mathbb{E}_{\boldsymbol{x},\boldsymbol{W}}[\varphi(\boldsymbol{x}) \otimes \varphi(\boldsymbol{x})]$

Problem settings: RFMs with the squared loss by SGD

Online SGD: one-pass, average output, adaptive step-size...

$$\boldsymbol{a}_t = \boldsymbol{a}_{t-1} + \gamma_t [y_t - \langle \boldsymbol{a}_{t-1}, \varphi(\boldsymbol{x}_t) \rangle] \varphi(\boldsymbol{x}_t), \qquad t = 1, 2, \ldots n \,.$$

Problem settings: RFMs with the squared loss by SGD

Online SGD: one-pass, average output, adaptive step-size...

$$\boldsymbol{a}_t = \boldsymbol{a}_{t-1} + \gamma_t [y_t - \langle \boldsymbol{a}_{t-1}, \varphi(\boldsymbol{x}_t) \rangle] \varphi(\boldsymbol{x}_t), \qquad t = 1, 2, \dots n\,.$$

- averaged output: $\bar{\boldsymbol{a}}_n := \frac{1}{n} \sum_{t=0}^{n-1} \boldsymbol{a}_t \Longrightarrow \bar{f}_n = \langle \varphi(\cdot), \bar{\boldsymbol{a}}_n \rangle$
- adaptive step-size: $\gamma_t := \gamma_0 t^{-\zeta}, \zeta \in [0, 1)$

Problem settings: RFMs with the squared loss by SGD

Online SGD: one-pass, average output, adaptive step-size...

$$\boldsymbol{a}_t = \boldsymbol{a}_{t-1} + \gamma_t [y_t - \langle \boldsymbol{a}_{t-1}, \varphi(\boldsymbol{x}_t) \rangle] \varphi(\boldsymbol{x}_t), \qquad t = 1, 2, \ldots n\,.$$

- averaged output: $\bar{\boldsymbol{a}}_n := \frac{1}{n} \sum_{t=0}^{n-1} \boldsymbol{a}_t \Longrightarrow \bar{f}_n = \langle \varphi(\cdot), \bar{\boldsymbol{a}}_n \rangle$
- adaptive step-size: $\gamma_t := \gamma_0 t^{-\zeta}, \zeta \in [0, 1)$

Averaged expected risk

- optimal solution: $f^* = \arg\min_{f \in \mathcal{H}} \|f - f_\rho\|^2_{L^2_{\rho_X}}$ with $\|f^*\|_{\mathcal{H}} < \infty$
- averaged excess risk: $\mathbb{E}\|\bar{f}_n - f^*\|^2_{L^2_{\rho_X}} = \mathbb{E}_{\boldsymbol{X}, \boldsymbol{W}, \boldsymbol{\varepsilon}} \langle \bar{f}_n - f^*, \Sigma_m (\bar{f}_n - f^*) \rangle$

Assumptions

Assumption (Basic assumptions)

- **non-asymptotic:** $\|\boldsymbol{x}\|_2^2 \leq \mathcal{O}(d)$, $\Sigma_d := \mathbb{E}_{\boldsymbol{x}}[\boldsymbol{x} \otimes \boldsymbol{x}]$ *with* $\|\Sigma_d\|_2 < \infty$
- **boundedness of f^*:** $\|f^*\|_{\mathcal{H}} < \infty$
- **activation function:** $\sigma(\cdot)$*: Lipschitz continuous*
- **label noise:** $\mathbb{E}(\varepsilon) = 0$ *and* $\mathbb{E}(\varepsilon^2) = \tau^2$

Assumptions

Assumption (Basic assumptions)

- **non-asymptotic:** $\|\boldsymbol{x}\|_2^2 \leq \mathcal{O}(d)$, $\Sigma_d := \mathbb{E}_{\boldsymbol{x}}[\boldsymbol{x} \otimes \boldsymbol{x}]$ *with* $\|\Sigma_d\|_2 < \infty$
- **boundedness of f^*:** $\|f^*\|_{\mathcal{H}} < \infty$
- **activation function:** $\sigma(\cdot)$*: Lipschitz continuous*
- **label noise:** $\mathbb{E}(\varepsilon) = 0$ *and* $\mathbb{E}(\varepsilon^2) = \tau^2$

Assumption (Fourth moment condition)

for any PSD operator A, we assume

$$\mathbb{E}_{\boldsymbol{W}}[\Sigma_m A \Sigma_m] \preccurlyeq r' \mathbb{E}_{\boldsymbol{W}}[\mathrm{Tr}(\Sigma_m A)\Sigma_m] \preccurlyeq r \mathrm{Tr}(\widetilde{\Sigma}_m A)\widetilde{\Sigma}_m \,.$$

Remark:

- the special case $A := I$ can be proved.
- holds for sub-Gaussian data.
- widely used in SGD analysis [8, 9, 10]

Main results: bias-variance decomposition

Define $\eta_t := f_t - f^*$, we have

$$\eta_t = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)](f_{t-1} - f^*) + \gamma_t \varepsilon_t \varphi(\boldsymbol{x}_t)\,,$$

Main results: bias-variance decomposition

Define $\eta_t := f_t - f^*$, we have

$$\eta_t = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)](f_{t-1} - f^*) + \gamma_t \varepsilon_t \varphi(\boldsymbol{x}_t)\,,$$

$$\eta_t^{\texttt{bias}} = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)]\eta_{t-1}^{\texttt{bias}}, \quad \eta_0^{\texttt{bias}} = f^*\,,$$

Main results: bias-variance decomposition

Define $\eta_t := f_t - f^*$, we have

$$\eta_t = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)](f_{t-1} - f^*) + \gamma_t \varepsilon_t \varphi(\boldsymbol{x}_t)\,,$$

$$\eta_t^{\texttt{bias}} = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)]\eta_{t-1}^{\texttt{bias}}, \quad \eta_0^{\texttt{bias}} = f^*\,,$$

$$\eta_t^{\texttt{var}} = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)]\eta_{t-1}^{\texttt{var}} + \gamma_t \varepsilon_t \varphi(\boldsymbol{x}_t), \quad \eta_0^{\texttt{var}} = 0\,.$$

Main results: bias-variance decomposition

Define $\eta_t := f_t - f^*$, we have

$$\eta_t = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)](f_{t-1} - f^*) + \gamma_t \varepsilon_t \varphi(\boldsymbol{x}_t)\,,$$

$$\eta_t^{\texttt{bias}} = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)]\eta_{t-1}^{\texttt{bias}}, \quad \eta_0^{\texttt{bias}} = f^*\,,$$

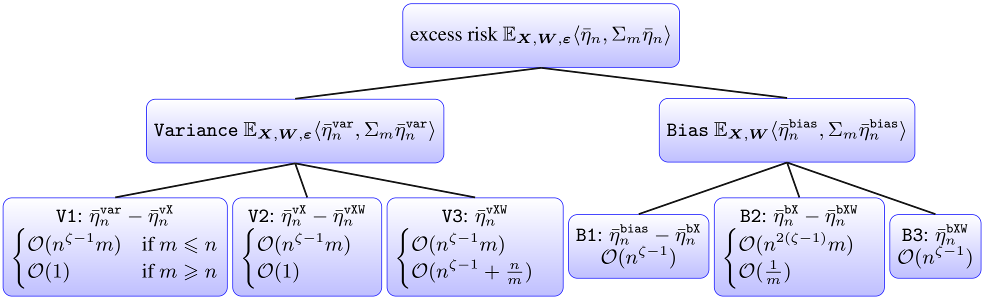
$$\eta_t^{\texttt{var}} = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)]\eta_{t-1}^{\texttt{var}} + \gamma_t \varepsilon_t \varphi(\boldsymbol{x}_t), \quad \eta_0^{\texttt{var}} = 0\,.$$

Theorem (Bias-variance decomposition)

Under the above-mentioned assumptions, if the step-size $\gamma_t := \gamma_0 t^{-\zeta}$ with $\zeta \in [0,1)$ satisfies $\gamma_0 < C$, we have

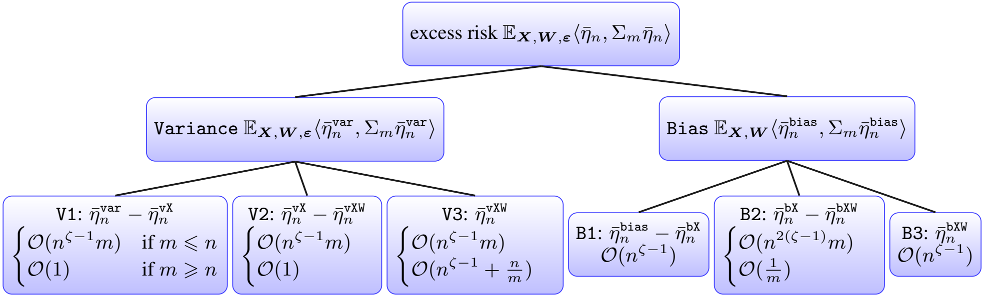
$$\mathbb{E}\|\bar{f}_n - f^*\|^2_{L^2_{\rho_X}} = \underbrace{\mathbb{E}_{\boldsymbol{X},\boldsymbol{W}}\langle \bar{\eta}_n^{\texttt{bias}}, \Sigma_m \bar{\eta}_n^{\texttt{bias}}\rangle}_{:=\texttt{Bias}} + \underbrace{\mathbb{E}_{\boldsymbol{X},\boldsymbol{W},\boldsymbol{\varepsilon}}\langle \bar{\eta}_n^{\texttt{var}}, \Sigma_m \bar{\eta}_n^{\texttt{var}}\rangle}_{:=\texttt{Variance}}\,.$$

Proof framework: randomness decoupling



$$\texttt{Bias}: \quad \eta_t^{\texttt{bias}} = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)]\eta_{t-1}^{\texttt{bias}}$$

Proof framework: randomness decoupling



$$\mathtt{Bias}: \quad \eta_t^{\mathtt{bias}} = [I - \gamma_t \varphi(\boldsymbol{x}_t) \otimes \varphi(\boldsymbol{x}_t)]\eta_{t-1}^{\mathtt{bias}}$$

Define "semi-stochastic" version: $\eta_t^{\mathtt{bX}} = (I - \gamma_t \Sigma_m)\eta_{t-1}^{\mathtt{bX}}, \quad \eta_t^{\mathtt{bXW}} = (I - \gamma_t \widetilde{\Sigma}_m)\eta_{t-1}^{\mathtt{bXW}},$

- $\mathtt{B1} := \mathbb{E}_{\boldsymbol{X},\boldsymbol{W}}\left[\langle \bar{\eta}_n^{\mathtt{bias}} - \bar{\eta}_n^{\mathtt{bX}}, \Sigma_m(\bar{\eta}_n^{\mathtt{bias}} - \bar{\eta}_n^{\mathtt{bX}})\rangle\right]$
- $\mathtt{B2} := \mathbb{E}_{\boldsymbol{W}}\left[\langle \bar{\eta}_n^{\mathtt{bX}} - \bar{\eta}_n^{\mathtt{bXW}}, \Sigma_m(\bar{\eta}_n^{\mathtt{bX}} - \bar{\eta}_n^{\mathtt{bXW}})\rangle\right]$
- $\mathtt{B3} := \langle \bar{\eta}_n^{\mathtt{bXW}}, \widetilde{\Sigma}_m \bar{\eta}_n^{\mathtt{bXW}} \rangle$

Proof framework: properties of covariance operators

Properties of $\widetilde{\Sigma}_m$

- the diagonal elements are the same $a := [\widetilde{\Sigma}_m]_{ii}, \forall i \in [m]$
- the non-diagonal elements are the same $b := [\widetilde{\Sigma}_m]_{ij}, \forall i,j \in [m], i \neq j$

$$\widetilde{\Sigma}_m = (a-b)\boldsymbol{I}_m + b\mathbf{1}\mathbf{1}^\top$$

- two distinct eigenvalues: $\widetilde{\lambda}_1 = a - b + bm \sim \mathcal{O}(1)$, $\widetilde{\lambda}_2 = \cdots = \widetilde{\lambda}_m = a - b \sim \mathcal{O}(1/m)$

Proof framework: properties of covariance operators

Properties of $\widetilde{\Sigma}_m$

- the diagonal elements are the same $a := [\widetilde{\Sigma}_m]_{ii}, \forall i \in [m]$
- the non-diagonal elements are the same $b := [\widetilde{\Sigma}_m]_{ij}, \forall i,j \in [m], i \neq j$

$$\widetilde{\Sigma}_m = (a-b)\boldsymbol{I}_m + b\mathbf{1}\mathbf{1}^\top$$

- two distinct eigenvalues: $\widetilde{\lambda}_1 = a - b + bm \sim \mathcal{O}(1)$, $\widetilde{\lambda}_2 = \cdots = \widetilde{\lambda}_m = a - b \sim \mathcal{O}(1/m)$

Example (ReLU activation)

- $(\widetilde{\Sigma}_m)_{ii} = \frac{1}{2md}\mathrm{Tr}(\Sigma_d)$
- $(\widetilde{\Sigma}_m)_{ij} = \frac{1}{2md\pi}\mathrm{Tr}(\Sigma_d)$

Proof framework: properties of covariance operators

Properties of $\widetilde{\Sigma}_m$

- the diagonal elements are the same $a := [\widetilde{\Sigma}_m]_{ii}, \forall i \in [m]$
- the non-diagonal elements are the same $b := [\widetilde{\Sigma}_m]_{ij}, \forall i, j \in [m], i \neq j$

$$\widetilde{\Sigma}_m = (a-b)\boldsymbol{I}_m + b\mathbf{1}\mathbf{1}^\top$$

- two distinct eigenvalues: $\widetilde{\lambda}_1 = a - b + bm \sim \mathcal{O}(1)$, $\widetilde{\lambda}_2 = \cdots = \widetilde{\lambda}_m = a - b \sim \mathcal{O}(1/m)$

Example (ReLU activation)

- $(\widetilde{\Sigma}_m)_{ii} = \frac{1}{2md}\mathrm{Tr}(\Sigma_d)$
- $(\widetilde{\Sigma}_m)_{ij} = \frac{1}{2md\pi}\mathrm{Tr}(\Sigma_d)$

sub-exponential random variables

$\|\Sigma_m\|_2$, $\|\Sigma_m - \widetilde{\Sigma}_m\|_2$, $\mathrm{Tr}(\Sigma_m)$, and $\left\|\widetilde{\Sigma}_m^{-1}\mathbb{E}_{\boldsymbol{W}}(\Sigma_m^2)\right\|_2$ with $\mathcal{O}(1)$ sub-exponential norm order

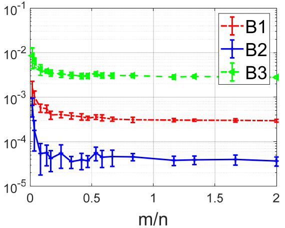
Main theorem

Theorem (Liu, Suykens, Volkan, NeurIPS 2022)

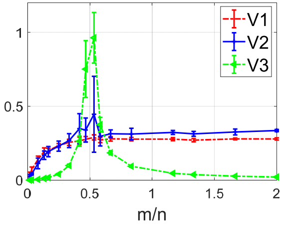
Under the above-mentioned assumptions, if the step-size $\gamma_t := \gamma_0 t^{-\zeta}$ *with* $\zeta \in [0,1)$ *satisfies* $\gamma_0 < C$*, we have*

$$\texttt{Bias} \lesssim \gamma_0 r' n^{\zeta-1} \|f^*\|^2 \sim \mathcal{O}\left(n^{\zeta-1}\right) .$$

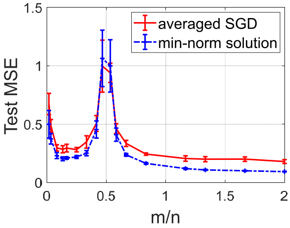
$$\texttt{Variance} \lesssim \gamma_0 r' \tau^2 \begin{cases} m n^{\zeta-1}, & \text{if } m \leqslant n \\ 1 + n^{\zeta-1} + \dfrac{n}{m}, & \text{if } m > n \end{cases}$$



(c) bias



(d) variance



(e) excess risk

Discussion

Constant step-size SGD doesn't hurt the convergence rate.

- under-parameterized regime (by taking $m = \mathcal{O}(\sqrt{n})$)

$$\mathbb{E}\|\bar{f}_n - f^*\|^2_{L^2_{\rho_X}} = \underbrace{\texttt{Bias}}_{\mathcal{O}(\frac{1}{n})} + \underbrace{\texttt{Variance}}_{\mathcal{O}(\frac{1}{\sqrt{n}})} \leq \mathcal{O}\left(\frac{1}{\sqrt{n}}\right),$$

  matches [11] (Carratino, Rudi, Rosasco, 2018) under one-pass, one-batch, SGD...[1]
- over-parameterized regime: matches [12] (Belkin, Hsu, Xu, 2020)

○ no lower bound: $\texttt{Bias} \leqslant 3(\texttt{B1} + \texttt{B2} + \texttt{B3})$ based on Minkowski inequality

[1] but the selection on step-size is different

Discussion

Constant step-size SGD doesn't hurt the convergence rate.

- under-parameterized regime (by taking $m = \mathcal{O}(\sqrt{n})$)

$$\mathbb{E}\|\bar{f}_n - f^*\|^2_{L^2_{\rho_X}} = \underbrace{\texttt{Bias}}_{\mathcal{O}(\frac{1}{n})} + \underbrace{\texttt{Variance}}_{\mathcal{O}(\frac{1}{\sqrt{n}})} \leq \mathcal{O}\left(\frac{1}{\sqrt{n}}\right),$$

  matches [11] (Carratino, Rudi, Rosasco, 2018) under one-pass, one-batch, SGD...[1]
- over-parameterized regime: matches [12] (Belkin, Hsu, Xu, 2020)

○ no lower bound: $\texttt{Bias} \leqslant 3(\texttt{B1} + \texttt{B2} + \texttt{B3})$ based on Minkowski inequality

Do you believe double descent?

[1] but the selection on step-size is different

What is the “right” model complexity?

- Complexity of a prediction rule, e.g.,
  - number of parameters
  - norm of parameter vector
  - norm of functions in RKHS

What is the "right" model complexity?

◦ Complexity of a prediction rule, e.g.,

- number of parameters
- norm of parameter vector
- norm of functions in RKHS

kernel methods to neural networks

- model complexity: from #params to norm constrained
- function space: from RKHS to ?

$$k(\boldsymbol{x}, \boldsymbol{x}') = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle_{\mathcal{H}}$$

What is the “right” model complexity?

- Complexity of a prediction rule, e.g.,
  - number of parameters
  - norm of parameter vector
  - norm of functions in RKHS

kernel methods to neural networks

- model complexity: from #params to norm constrained
- function space: from RKHS to ?

$$k(\boldsymbol{x}, \boldsymbol{x}') = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle_{\mathcal{H}}$$

- not data-adaptive
- RKHS is too small: curse of dimensionality [7, 13, 14]

From RKHS to Barron space

◦ RKHS of RFMs:

$$\hat{k}_m(\boldsymbol{x}, \boldsymbol{x}') = \frac{1}{m} \sum_{i=1}^{m} \phi(\boldsymbol{x}, \boldsymbol{w}_i)\phi(\boldsymbol{x}', \boldsymbol{w}_i) \;\; {\color{red}\rightarrow} \;\; k_\mu(\boldsymbol{x}, \boldsymbol{x}') = \int_{\mathcal{W}} \phi(\boldsymbol{x}, \boldsymbol{w})\phi(\boldsymbol{x}', \boldsymbol{w}) \mathrm{d}\mu(\boldsymbol{w})$$

From RKHS to Barron space

- RKHS of RFMs:

$$\hat{k}_m(\boldsymbol{x}, \boldsymbol{x}') = \frac{1}{m} \sum_{i=1}^{m} \phi(\boldsymbol{x}, \boldsymbol{w}_i)\phi(\boldsymbol{x}', \boldsymbol{w}_i) \rightarrow k_\mu(\boldsymbol{x}, \boldsymbol{x}') = \int_{\mathcal{W}} \phi(\boldsymbol{x}, \boldsymbol{w})\phi(\boldsymbol{x}', \boldsymbol{w}) \mathrm{d}\mu(\boldsymbol{w})$$

Definition (Barron space [15] (E, Ma, Wu, 2021))

$$\mathcal{B} = \cup_{\mu \in \mathcal{P}(\mathcal{W})} \mathcal{H}_{k_\mu}, \quad \|f\|_{\mathcal{B}} = \inf_{\mu \in \mathcal{P}(\mathcal{W})} \|f\|_{\mathcal{H}_{k_\mu}}$$

From RKHS to Barron space

∘ RKHS of RFMs:

$$\hat{k}_m(\boldsymbol{x}, \boldsymbol{x}') = \frac{1}{m}\sum_{i=1}^{m} \phi(\boldsymbol{x}, \boldsymbol{w}_i)\phi(\boldsymbol{x}', \boldsymbol{w}_i) \;\rightarrow\; k_\mu(\boldsymbol{x}, \boldsymbol{x}') = \int_{\mathcal{W}} \phi(\boldsymbol{x}, \boldsymbol{w})\phi(\boldsymbol{x}', \boldsymbol{w})\mathrm{d}\mu(\boldsymbol{w})$$

Definition (Barron space [15] (E, Ma, Wu, 2021))

$$\mathcal{B} = \cup_{\mu \in \mathcal{P}(\mathcal{W})} \mathcal{H}_{k_\mu}, \quad \|f\|_{\mathcal{B}} = \inf_{\mu \in \mathcal{P}(\mathcal{W})} \|f\|_{\mathcal{H}_{k_\mu}}$$

Remark: ∘ Two-layer neural networks: data-adaptive kernel

From RKHS to Barron space

∘ RKHS of RFMs:

$$\hat{k}_m(\boldsymbol{x}, \boldsymbol{x}') = \frac{1}{m}\sum_{i=1}^{m}\phi(\boldsymbol{x}, \boldsymbol{w}_i)\phi(\boldsymbol{x}', \boldsymbol{w}_i) \;\to\; k_\mu(\boldsymbol{x}, \boldsymbol{x}') = \int_{\mathcal{W}}\phi(\boldsymbol{x}, \boldsymbol{w})\phi(\boldsymbol{x}', \boldsymbol{w})\mathrm{d}\mu(\boldsymbol{w})$$

Definition (Barron space [15] (E, Ma, Wu, 2021))

$$\mathcal{B} = \cup_{\mu\in\mathcal{P}(\mathcal{W})}\mathcal{H}_{k_\mu}\,, \quad \|f\|_{\mathcal{B}} = \inf_{\mu\in\mathcal{P}(\mathcal{W})}\|f\|_{\mathcal{H}_{k_\mu}}$$

Remark: ∘ Two-layer neural networks: data-adaptive kernel
∘ equivalent to path norm $\|\boldsymbol{\Theta}\|_{\mathcal{P}} := \frac{1}{m}\sum_{k=1}^{m}|a_k|\|\boldsymbol{w}_k\|_1$

From RKHS to Barron space

∘ RKHS of RFMs:

$$\hat{k}_m(\boldsymbol{x}, \boldsymbol{x}') = \frac{1}{m} \sum_{i=1}^{m} \phi(\boldsymbol{x}, \boldsymbol{w}_i)\phi(\boldsymbol{x}', \boldsymbol{w}_i) \ \rightarrow \ k_\mu(\boldsymbol{x}, \boldsymbol{x}') = \int_{\mathcal{W}} \phi(\boldsymbol{x}, \boldsymbol{w})\phi(\boldsymbol{x}', \boldsymbol{w}) \mathrm{d}\mu(\boldsymbol{w})$$

Definition (Barron space [15] (E, Ma, Wu, 2021))

$$\mathcal{B} = \cup_{\mu \in \mathcal{P}(\mathcal{W})} \mathcal{H}_{k_\mu}, \quad \|f\|_{\mathcal{B}} = \inf_{\mu \in \mathcal{P}(\mathcal{W})} \|f\|_{\mathcal{H}_{k_\mu}}$$

Remark: ∘ Two-layer neural networks: data-adaptive kernel
∘ equivalent to path norm $\|\boldsymbol{\Theta}\|_{\mathcal{P}} := \frac{1}{m} \sum_{k=1}^{m} |a_k| \|\boldsymbol{w}_k\|_1$
∘ parameter space vs. measure space
e.g., [7] (Bach, 2017), [16] (Bartolucci, Vito, Rosasco, Vigogna, 2022).

Our results: Refined analyses in Barron spaces[2]

For the class of two-layer neural networks $\mathcal{F}_m$

$$\boldsymbol{\theta}^\star = \arg\min_{f_{\boldsymbol{\theta}} \in \mathcal{F}_m} \frac{1}{n} \sum_{i=1}^{n} (y_i - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))^2 + \lambda \|\boldsymbol{\theta}\|_{\mathcal{P}} \,.$$

[2] Fanghui Liu, Leello Dadi, Volkan Cevher. Learning with two-layer, norm-constrained, over-parameterized neural networks. JMLR (under the cond-round review)

Our results: Refined analyses in Barron spaces[2]

For the class of two-layer neural networks $\mathcal{F}_m$

$$\boldsymbol{\theta}^\star = \operatorname*{arg\,min}_{f_{\boldsymbol{\theta}} \in \mathcal{F}_m} \frac{1}{n} \sum_{i=1}^{n} (y_i - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))^2 + \lambda \|\boldsymbol{\theta}\|_{\mathcal{P}} \,.$$

Theorem (Informal)

Under proper assumptions, for two-layer over-parameterized neural networks, learning in Barron spaces leads to

$$\left\| f_{\boldsymbol{\theta}^\star} - f_\rho \right\|^2_{L^2_{\rho_X}} \lesssim \lambda + \frac{1}{m} + d^2 n^{-\frac{d+2}{2d+2}} \qquad w.h.p.$$

[2] Fanghui Liu, Leello Dadi, Volkan Cevher. Learning with two-layer, norm-constrained, over-parameterized neural networks. JMLR (under the cond-round review)

Our results: Refined analyses in Barron spaces[2]

For the class of two-layer neural networks $\mathcal{F}_m$

$$\boldsymbol{\theta}^\star = \operatorname*{arg\,min}_{f_{\boldsymbol{\theta}} \in \mathcal{F}_m} \frac{1}{n} \sum_{i=1}^n (y_i - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))^2 + \lambda \|\boldsymbol{\theta}\|_{\mathcal{P}} \,.$$

Theorem (Informal)

Under proper assumptions, for two-layer over-parameterized neural networks, learning in Barron spaces leads to

$$\left\| f_{\boldsymbol{\theta}^\star} - f_\rho \right\|^2_{L^2_{\rho_X}} \lesssim \lambda + \frac{1}{m} + d^2 n^{-\frac{d+2}{2d+2}} \qquad w.h.p.$$

Remark:

- [17] (Siegel, Xu, 2022) on metric entropy

$$\epsilon^{-\frac{2d}{d+3}} \lesssim_d \log \mathcal{N}_2(\mathcal{G}_1, \epsilon) \lesssim_d \epsilon^{-\frac{2d}{d+3}} \,.$$

[2] Fanghui Liu, Leello Dadi, Volkan Cevher. Learning with two-layer, norm-constrained, over-parameterized neural networks. JMLR (under the cond-round review)

Our results: Refined analyses in Barron spaces[2]

For the class of two-layer neural networks $\mathcal{F}_m$

$$\boldsymbol{\theta}^\star = \operatorname*{arg\,min}_{f_{\boldsymbol{\theta}} \in \mathcal{F}_m} \frac{1}{n} \sum_{i=1}^{n} (y_i - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))^2 + \lambda \|\boldsymbol{\theta}\|_{\mathcal{P}} \,.$$

Theorem (Informal)

Under proper assumptions, for two-layer over-parameterized neural networks, learning in Barron spaces leads to

$$\left\| f_{\boldsymbol{\theta}^\star} - f_\rho \right\|_{L^2_{\rho_X}}^2 \lesssim \lambda + \frac{1}{m} + d^2 n^{-\frac{d+2}{2d+2}} \qquad w.h.p.$$

Remark:

- [17] (Siegel, Xu, 2022) on metric entropy

$$\epsilon^{-\frac{2d}{d+3}} \lesssim_d \log \mathcal{N}_2(\mathcal{G}_1, \epsilon) \cancel{\lesssim_d \epsilon^{-\frac{2d}{d+3}}} \leqslant 6144 d^5 \epsilon^{-\frac{2d}{d+2}} \quad \text{[Ours]}$$

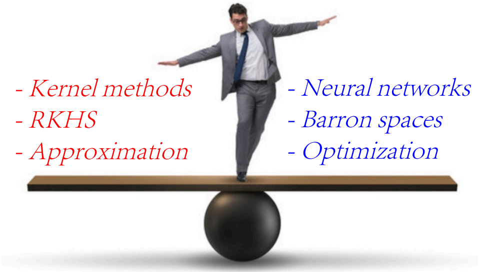
[2] Fanghui Liu, Leello Dadi, Volkan Cevher. Learning with two-layer, norm-constrained, over-parameterized neural networks. JMLR (under the cond-round review)

Optimization in Barron spaces is difficult: curse of dimensionality!

	approximation	generalization	optimization
RKHS	CoD	$\mathcal{O}(n^{-\frac{1}{d}})$	-
Barron spaces	$\mathcal{O}(m^{-\frac{2d}{d+3}})$	$\mathcal{O}(n^{-\frac{d+3}{2d+3}})$?	CoD

Optimization in Barron spaces is difficult: curse of dimensionality!

	approximation	generalization	optimization
RKHS	CoD	$\mathcal{O}(n^{-\frac{1}{d}})$	-
Barron spaces	$\mathcal{O}(m^{-\frac{2d}{d+3}})$	$\mathcal{O}(n^{-\frac{d+3}{2d+3}})$?	CoD



What is the suitable function space of NNs, both statistically and computationally efficient?

What is the suitable function space of NNs, both statistically and computationally efficient?

- *Random Features for Kernel Approximation: A Survey on Algorithms, Theory, and Beyond.* (Liu, Huang, Chen, Suykens, TPAMI2021).
- IEEE ICASSP 2023 Tutorial - “Neural networks: the good, the bad, and the ugly”
- CVPR 2023 Tutorial - “Deep learning theory for computer vision”

Thanks for your attention!

Q & A

my homepage `www.lfhsgre.org` for more information!

References I

[1] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever.
Deep double descent: Where bigger models and more data hurt.
In *International Conference on Learning Representations*, 2019.
(Cited on pages 3 and 4.)

[2] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal.
Reconciling modern machine-learning practice and the classical bias–variance trade-off.
the National Academy of Sciences, 116(32):15849–15854, 2019.
(Cited on pages 3 and 4.)

[3] Ali Rahimi and Benjamin Recht.
Random features for large-scale kernel machines.
In *Advances in Neural Information Processing Systems*, pages 1177–1184, 2007.
(Cited on pages 5 and 6.)

[4] Song Mei and Andrea Montanari.
The generalization error of random features regression: Precise asymptotics and the double descent curve.
Communications on Pure and Applied Mathematics, 75(4):667–766, 2022.
(Cited on pages 7, 8, 9, and 10.)

References II

[5] Behrooz Ghorbani, Song Mei, Theodor Misiakiewicz, and Andrea Montanari.
Linearized two-layers neural networks in high dimension.
Annals of Statistics, 49(2):1029–1054, 2021.
(Cited on pages 11, 12, 13, and 14.)

[6] Noureddine El Karoui.
The spectrum of kernel random matrices.
Annals of Statistics, 38(1):1–50, 2010.
(Cited on pages 11, 12, 13, and 14.)

[7] Francis Bach.
Breaking the curse of dimensionality with convex neural networks.
Journal of Machine Learning Research, 18(1):629–681, 2017.
(Cited on pages 11, 12, 13, 14, 37, 38, 39, 40, 41, 42, 43, and 44.)

[8] Francis Bach and Eric Moulines.
Non-strongly-convex smooth stochastic approximation with convergence rate $o(1/n)$.
Advances in Neural Information Processing Systems, 26:773–781, 2013.
(Cited on pages 23 and 24.)

References III

[9] Prateek Jain, Sham M Kakade, Rahul Kidambi, Praneeth Netrapalli, Venkata Krishna Pillutla, and Aaron Sidford.
A markov chain theory approach to characterizing the minimax optimality of stochastic gradient descent (for least squares).
arXiv preprint arXiv:1710.09430, 2017.
(Cited on pages 23 and 24.)

[10] Difan Zou, Jingfeng Wu, Vladimir Braverman, Quanquan Gu, and Sham M Kakade.
Benign overfitting of constant-stepsize sgd for linear regression.
In *Conference on Learning Theory*, 2021.
(Cited on pages 23 and 24.)

[11] Luigi Carratino, Alessandro Rudi, and Lorenzo Rosasco.
Learning with SGD and random features.
In *Advances in Neural Information Processing Systems*, pages 10212–10223, 2018.
(Cited on pages 35 and 36.)

[12] Mikhail Belkin, Daniel Hsu, and Ji Xu.
Two models of double descent for weak features.
SIAM Journal on Mathematics of Data Science, 2(4):1167–1180, 2020.
(Cited on pages 35 and 36.)

References IV

[13] Gilad Yehudai and Ohad Shamir.
On the power and limitations of random features for understanding neural networks.
In *Advances in Neural Information Processing Systems*, pages 6594–6604, 2019.
(Cited on pages 37, 38, and 39.)

[14] Michael Celentano, Theodor Misiakiewicz, and Andrea Montanari.
Minimum complexity interpolation in random features models.
arXiv preprint arXiv:2103.15996, 2021.
(Cited on pages 37, 38, and 39.)

[15] Weinan E, Chao Ma, and Lei Wu.
The barron space and the flow-induced function spaces for neural network models.
Constructive Approximation, pages 1–38, 2021.
(Cited on pages 40, 41, 42, 43, and 44.)

[16] Francesca Bartolucci, Ernesto De Vito, Lorenzo Rosasco, and Stefano Vigogna.
Understanding neural networks with reproducing kernel Banach spaces.
Applied and Computational Harmonic Analysis, 2023.
(Cited on pages 40, 41, 42, 43, and 44.)

References V

[17] Jonathan W Siegel and Jinchao Xu.
Sharp bounds on the approximation rates, metric entropy, and n-widths of shallow neural networks.
arXiv preprint arXiv:2101.12365, 2021.
(Cited on pages 45, 46, 47, and 48.)