The Mathematical Theory of Neural Network-Based Machine Learning

Weinan E

Joint work with:

Chao Ma, Lei Wu, Qingcan Wang

Jiequn Han, Arnulf Jentzen, Qianxiao Li.

Outline

Outline

Supervised learning: Approximating functions using samples

- Object of interest: (f^*, μ), where $f^* : \mathbb{R}^d \to \mathbb{R}^1$, μ is a prob measure on $\mathbb{R}^d$.
- Given a set of samples from μ, $\{\boldsymbol{x}_j\}_{j=1}^n$, and $\{y_j = f^*(\boldsymbol{x}_j)\}_{j=1}^n$
- Task: Approximate f^* using $S = \{(\boldsymbol{x}_j, y_j)\}_{j=1}^n$.

- Strategy: Construct some "hypothesis space" (space of functions) $\mathcal{H}_m$ ($m \sim$ the dimension of $\mathcal{H}_m$). Minimize the "empirical risk":

$$\hat{\mathcal{R}}_n(\theta) = \frac{1}{n}\sum_j (f(\boldsymbol{x}_j) - y_j)^2 = \frac{1}{n}\sum_j (f(\boldsymbol{x}_j) - f^*(\boldsymbol{x}_j))^2$$

- Remark: What we **really** want to minimize is the "population risk":

$$\mathcal{R}(\theta) = \mathbb{E}(f(\boldsymbol{x}) - f^*(\boldsymbol{x}))^2 = \int_{\mathbb{R}^d} (f(\boldsymbol{x}) - f^*(\boldsymbol{x}))^2 d\mu$$

- **Key words: High dimension and over-parametrization**

Issues that we would like to understand

Basic questions:

- high dimensionality
- models are highly over-parametrized, classical machine learning theory would suggest overfitting
- models are non-convex, yet simple gradient algorithms seem to work (compare with structural optimization in material science and chemistry)

Advanced questions:

- Why deep networks seem to perform better than shallow ones?
- Why stochastic gradient descent seems to perform better than gradient descent?
- Lots of other issues, mysteries, e.g. batch normalization, dropout, initialization, ...

Over-parametrization and curse of dimensionality

$$\mathcal{H}_m = \{f = \sum_{k=1}^{m} a_k \phi_k\}$$

$m > n$, $\{\phi_k\}$ are linearly independent functions.

- Given a data set $\{\boldsymbol{x}_j, y_j; j = 1, \cdots n\}$, one can interpolate the data

$$G\boldsymbol{a} = \boldsymbol{y}, \quad G = (\phi_k(\boldsymbol{x}_j))$$

- Yet there is curse of dimensionality

$$\sup_{\|f\|_{\mathcal{B}_1} \leq 1} \inf_{h \in \mathcal{H}_m} \|f - h\|_{L^2(D_0)} \geq \frac{C}{dm^{1/d}}$$

Approximating functions in high dimensions

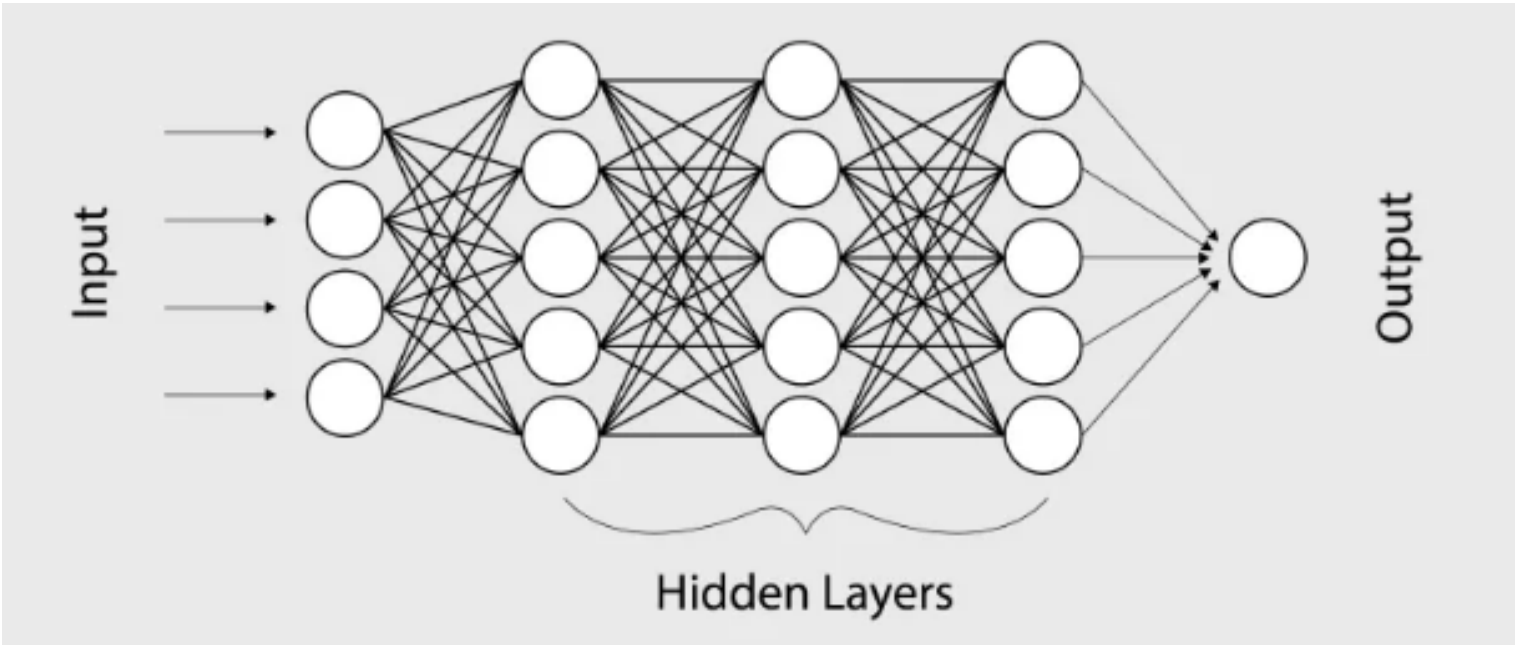
Given (f^*, μ), where $f^* : \mathbb{R}^d \to \mathbb{R}^1$, μ is a prob measure on $\mathbb{R}^d$.

Minimize is the "population risk" over "hypothesis space":

$$L(\theta) = \mathbb{E}(f(\boldsymbol{x}) - f^*(\boldsymbol{x}))^2 = \int_{\mathbb{R}^d} (f(\boldsymbol{x}) - f^*(\boldsymbol{x}))^2 d\mu$$

How do we choose the hypothesis space?

- linear regression: $f(\boldsymbol{x}) = \beta \cdot \boldsymbol{x} + \beta_0$
- generalized linear models: $f(\boldsymbol{x}) = \sum_{k=1}^{m} c_k \phi_k(\boldsymbol{x})$, where $\{\phi_k\}$ are linearly independent functions.
- two-layer neural networks: $f(\boldsymbol{x}) = \sum_k a_k \sigma(\boldsymbol{b}_k \cdot \boldsymbol{x} + c_k)$, where σ is some nonlinear function, e.g. $\sigma(z) = \max(z, 0)$.
- deep neural networks (DNN) : compositions of functions of the form above.



$$f(\boldsymbol{x}, \theta) = \boldsymbol{W}_L \sigma \circ (\boldsymbol{W}_{L-1} \sigma \circ (\cdots \sigma \circ (\boldsymbol{W}_0 \boldsymbol{x}))), \quad \theta = (\boldsymbol{W}_0, \boldsymbol{W}_1, \cdots, \boldsymbol{W}_L)$$

σ is a scalar function.

- $\sigma(x) = \max(x, 0)$, the ReLU (rectified linear units) function.
- $\sigma(x) = (1 + e^{-x})^{-1}$, the "sigmoid function".
- $\sigma(x) = \cos(x)$

"$\circ$" means acting on each components, the $\boldsymbol{W}$'s are matrices.

Why neural networks?

Difference between linear and nonlinear approximations:

- Linear: $f(\boldsymbol{x}) \approx f_m(\boldsymbol{x}) = \sum^m a_{\boldsymbol{k}} \cos(2\pi \boldsymbol{k} \cdot \boldsymbol{x}), \boldsymbol{x} \in [0,1]^d$

$$\inf \|f - f_m\|_2 \geq C(f) m^{-1/d}$$

"Curse of dimensionality": number of parameters needed goes up exponentially fast as a function of the accuracy requirement.

- Nonlinear: $f(\boldsymbol{x}) \approx f_m(\boldsymbol{x}) = \sum^m a_{\boldsymbol{k}} \cos(2\pi \boldsymbol{b}_k \cdot \boldsymbol{x}), \boldsymbol{x} \in [0,1]^d$ (two-layer neural network, with $\cos(x)$ as the activation function).

$$\inf \|f - f_m\|_2 \leq C(f) m^{-1/2}$$

This is the best one can hope for.

Exploding and vanishing gradients

$$f(\boldsymbol{x}, \theta) = \boldsymbol{W}_L \sigma \circ (\boldsymbol{W}_{L-1} \sigma \circ (\cdots \sigma \circ (\boldsymbol{W}_0 \boldsymbol{x}))), \quad \theta = (\boldsymbol{W}_0, \boldsymbol{W}_1, \cdots, \boldsymbol{W}_L)$$

$$\text{“}\nabla_\theta f'' = \boldsymbol{W}_L \cdot \boldsymbol{W}_{L-1} \cdots \boldsymbol{W}_0 \sim \kappa^L, \quad L >> 1$$

exploding or vanishing gradients problem (see Hanin (2018)).

Solution: Using residual networks (He et al. (2016))

$$\begin{aligned} \boldsymbol{z}_{0,L}(\boldsymbol{x}) &= \boldsymbol{V}\boldsymbol{x}, \\ \boldsymbol{z}_{l+1,L}(\boldsymbol{x}) &= z_{l,L}(\boldsymbol{x}) + \frac{1}{L}\boldsymbol{U}_l \sigma \circ (\boldsymbol{W}_l z_{l,L}(\boldsymbol{x})), \quad l = 0, 1, \cdots, L-1 \end{aligned}$$

$$f(\boldsymbol{x}, \theta) = \alpha \cdot \boldsymbol{z}_{L,L}(\boldsymbol{x})$$

Dynamical system viewpoint to deep learning

Constructing nonlinear approximations through the **flow map** of a dynamical system (E (2017, Comm Math Stats), Chen et al (NeurIPS 2018, "Neural ODE")):

$$\frac{d\boldsymbol{z}(\boldsymbol{x},t)}{dt} = \boldsymbol{F}(\boldsymbol{z}(\boldsymbol{x},t)), \quad \boldsymbol{z}(0,\boldsymbol{x}) = \boldsymbol{V}\boldsymbol{x}$$

The flow map $\boldsymbol{x} \to \boldsymbol{z}(\boldsymbol{x},1)$ is a nonlinear mapping.

Simplest choice of (nonlinear) $\boldsymbol{F}$: $\boldsymbol{F}(\boldsymbol{z};\boldsymbol{U},\boldsymbol{W}) = \boldsymbol{U}\sigma \circ (\boldsymbol{W}\boldsymbol{z})$.

Choose the optimal $\boldsymbol{U}, \boldsymbol{W}(\cdot), \alpha$ to approximate f^* by

$$f^*(\boldsymbol{x}) \sim \alpha \cdot \boldsymbol{z}(\boldsymbol{x},1)$$

Control theory viewpoint to deep learning

LeCun (1988), E, Han and Q. Li (2018)

$$\frac{d\boldsymbol{z}(\boldsymbol{x},t)}{dt} = \boldsymbol{F}(\boldsymbol{z}(\boldsymbol{x},t);\boldsymbol{U}(t),\boldsymbol{W}(t)), \quad \boldsymbol{z}(\boldsymbol{x},0) = \boldsymbol{V}\boldsymbol{x}$$

$$I(\alpha,\boldsymbol{U},\boldsymbol{W}) = \int_{\mathbb{R}^d} \left((\alpha \cdot \boldsymbol{z}(\boldsymbol{x},1) - f^*(\boldsymbol{x}))^2 + \lambda_n \int_0^1 L(\boldsymbol{z}(\boldsymbol{x},1);\boldsymbol{U}(t),\boldsymbol{W}(t))dt \right) d\mu(\boldsymbol{x})$$

In practice, we have

$$\frac{d\boldsymbol{z}_j(t)}{dt} = \boldsymbol{F}(\boldsymbol{z}_j(t);\boldsymbol{U}(t),\boldsymbol{W}(t)), \quad \boldsymbol{z}_j = \boldsymbol{V}\boldsymbol{x}_j, \cdots, j = 1, \cdots, n$$

$$I_n(\alpha,\boldsymbol{U},\boldsymbol{W}) = \frac{1}{n}\sum_j \left((\alpha \cdot \boldsymbol{z}_j(1) - f^*(\boldsymbol{x}_j))^2 + \lambda_n \int_0^1 L(\boldsymbol{z}_j(t);\boldsymbol{U}(t),\boldsymbol{W}(t))dt \right)$$

Neural network models used in practice can be viewed as a discrete (in time) analog.

Maximum principle-based training algorithms: Q. Li et al (2017, 2018).

Classical numerical analysis (approximation theory)

- Define a "well-posed" math model (the hypothesis space, the loss function, etc)
  - splines: hypothesis space $= C^1$ piecewise cubic polynomials the data

$$I_n(f) = \frac{1}{n}\sum_{j=1}^{n}(f(x_j) - y_j)^2 + \lambda \int |D^2 f(x)|^2 dx$$

  - finite elements: hypothesis space $= C^0$ piecewise polynomials
- Identify the right function spaces, e.g. Sobolev/Besov spaces
  - direct and inverse approximation theorem (Bernstein and Jackson type theorems): f can be approximated by trig polynomials in L^2 to order s iff $f \in H^s$, $\|f\|_{H^s}^2 = \sum_{k=0}^{s} \|\nabla^k f\|_{L^2}^2$.
  - functions of interest are in the right spaces (PDE theory, real analysis, etc).
- Optimal error estimates
  - *A priori* estimates (for piecewise linear finite elements, $\alpha = 1/d, s = 2$)

$$\|f_m - f^*\|_{H^1} \leq Cm^{-\alpha}\|f^*\|_{H^s}$$

  - *A posteriori* estimates (say in finite elements):

$$\|f_m - f^*\|_{H^1} \leq Cm^{-\alpha}\|f_m\|_h$$

We will adopt a similar strategy, but aim for high dimensions.

Another benchmark: High dimensional integration

Monte Carlo: $X = [0,1]^d, \{\boldsymbol{x}_j, j = 1, \cdots, n\}$ is uniformly distributed in X.

$$I(g) = \int_X g(\boldsymbol{x}) d\mu, \quad I_n(g) = \frac{1}{n} \sum_j g(\boldsymbol{x}_j)$$

$$\mathbb{E}(I(g) - I_n(g))^2 = \frac{1}{n} \mathsf{Var}(g)$$

$\mathsf{Var}(g) = \int_X g^2(\boldsymbol{x}) d\boldsymbol{x} - (\int_X g(\boldsymbol{x}) d\boldsymbol{x})^2$

The $O(1/\sqrt{n})$ rate is the best we can hope for.

However, $\mathsf{Var}(g)$ can be very large in high dimension. That's why variance reduction is important!

We only know the values of the target function on a sample set

Let $\{(\boldsymbol{x}_i, y_i = f^*(\boldsymbol{x}_i)), i = 1, \cdots, n\}$. Work with the **empirical risk**

$$\hat{\mathcal{R}}_n(\theta) = \frac{1}{n}\sum_i (f(\boldsymbol{x}_i, \theta) - f^*(\boldsymbol{x}_i))^2, \quad \hat{\theta} = \text{argmin } \hat{\mathcal{R}}_n(\theta)$$

We are interested in an estimate of the **population risk** ("**generalization error**"):

$$\mathcal{R}(\hat{\theta}) = \mathbb{E}(f(\boldsymbol{x}, \hat{\theta}) - f^*(\boldsymbol{x}))^2$$

Difficulty:

- $\hat{\theta}$ is far from being unique when $m > n$ (over-parametrized regime).

  Yaim Cooper: Such $\hat{\theta}$'s form a $m - n$ dimensional (smooth) manifold.
- $\hat{\theta}$ is highly correlated with the data set S

  Study the worse case situation in the hypothesis space.

Expect the optimal error rate to be $O(1/m) + O(1/\sqrt{n})$, both are Monte Carlo rates.

Estimating the generalization gap

"Generalization gap" $= \hat{\mathcal{R}}(\hat{\theta}) - \hat{\mathcal{R}}_n(\hat{\theta}) = I(g) - I_n(g), \quad g(\boldsymbol{x}) = (f(\boldsymbol{x}, \theta) - f^*(\boldsymbol{x}))^2$

$$I(g) = \int_{X=[-1,1]^d} g(\boldsymbol{x}) d\mu, \quad I_n(g) = \frac{1}{n} \sum_j g(\boldsymbol{x}_j)$$

- For fixed $g = h$, we have
$$|I(h) - I_n(h)| \sim \frac{1}{\sqrt{n}}$$
- For Lipschitz functions (Wasserstein distance)
$$\sup_{\|h\|_{Lip} \leq 1} |I(h) - I_n(h)| \sim \frac{1}{n^{1/d}}$$
- For functions in Barron space, to be defined later
$$\sup_{\|h\|_{\mathcal{B}} \leq 1} |I(h) - I_n(h)| \sim \frac{1}{\sqrt{n}}$$

Rademacher complexity

Let $\mathcal{H}$ be a set of functions, and $S = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n)$ be a set of data points. Then, the Rademacher complexity of $\mathcal{H}$ with respect to S is defined as

$$\hat{R}_S(\mathcal{H}) = \frac{1}{n}\mathbb{E}_\xi \left[\sup_{h \in \mathcal{H}} \sum_{i=1}^{n} \xi_i h(\boldsymbol{x}_i) \right],$$

where $\{\xi_i\}_{i=1}^n$ are i.i.d. random variables taking values ± 1 with equal probability.

Theorem (Rademacher complexity and the generalization gap)

Given a function class $\mathcal{H}$, for any $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the random samples $S = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n)$,

$$\sup_{h \in \mathcal{H}} \left| \mathbb{E}_{\boldsymbol{x}} [h(\boldsymbol{x})] - \frac{1}{n} \sum_{i=1}^{n} h(\boldsymbol{x}_i) \right| \leq 2\hat{R}_S(\mathcal{H}) + \sup_{h \in \mathcal{H}} \|h\|_\infty \sqrt{\frac{\log(2/\delta)}{2n}}.$$

$$\sup_{h \in \mathcal{H}} \left| \mathbb{E}_{\boldsymbol{x}} [h(\boldsymbol{x})] - \frac{1}{n} \sum_{i=1}^{n} h(\boldsymbol{x}_i) \right| \geq \frac{1}{2}\hat{R}_S(\mathcal{H}) - \sup_{h \in \mathcal{H}} \|h\|_\infty \sqrt{\frac{\log(2/\delta)}{2n}}.$$

- If $\mathcal{H}$ contains a single function, then $\hat{R}_S(\mathcal{H}) \sim O(1/\sqrt{n})$
- If $\mathcal{H}$ contains functions that can fit any random values on S, then $\hat{R}_S(\mathcal{H}) \sim O(1)$
- If $\mathcal{H} =$ unit ball in Barron space: $\hat{R}_S(\mathcal{H}) \sim O(1/\sqrt{n})$
- If $\mathcal{H} =$ unit ball in Lipschitz space: $\hat{R}_S(\mathcal{H}) \sim O(1/n^{1/d})$
- If $\mathcal{H} =$ unit ball in C^0: $\hat{R}_S(\mathcal{H}) \sim O(1)$

Want: Large (function) space but low complexity: $\hat{R}_S(\mathcal{H}) \sim O(1/\sqrt{n})$ (related to Donsker spaces)

Two types of machine learning models

(1). Models that suffer from the curse of dimensionality:

$$\text{generalization error} = O(m^{-\alpha/d} + n^{-\beta/d})$$

- piecewise polynomial approximation
- wavelets with fixed wavelet basis

(2). Models that don't suffer from the curse of dimensionality:

$$\text{generalization error} = O(\gamma_1(f^*)/m + \gamma_2(f^*)/\sqrt{n})$$

- random feature models: $\{\phi(\cdot, \omega), \omega \in \Omega\}$ is the set of "features". Given any realization $\{\omega_j\}_{j=1}^m$, i.i.d. with distribution π, $\mathcal{H}_m(\{\omega_j\}) = \{f_m(\boldsymbol{x}, \boldsymbol{a}) = \frac{1}{m}\sum_{j=1}^m a_j \phi(\boldsymbol{x}; \omega_j).\}$.
- two layer neural networks $\mathcal{H}_m = \{\frac{1}{m}\sum_{j=1}^m a_j \sigma(\boldsymbol{b}_j^T \boldsymbol{x} + c_j)\}$
- residual neural networks $\mathcal{H}_L = \{f(\cdot, \theta) = \alpha \cdot \boldsymbol{z}_{L,L}(\cdot)\}$

$$\boldsymbol{z}_{l+1,L}(\boldsymbol{x}) = z_{l,L}(\boldsymbol{x}) + \frac{1}{L}\boldsymbol{U}_l \sigma \circ (\boldsymbol{W}_l z_{l,L}(\boldsymbol{x})), \quad \boldsymbol{z}_{0,L}(\boldsymbol{x}) = \boldsymbol{V}\boldsymbol{x}$$

Outline

Reproducing kernel Hilbert spaces

A symmetric function $k(\cdot,\cdot)$ is positive definite, if for any $\{\boldsymbol{x}_i\}_{i=1}^n$, the matrix $K \in \mathbb{R}^{n\times n}, K = (K_{i,j}), K_{i,j} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, is symmetric positive definite (SPD).

Reproducing kernel Hilbert space (RKHS)

$\mathcal{H}_k =$ the completion of $\{\sum_{i=1}^m \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x})\}$ with respect to the inner product given by

$$\langle \sum_i \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}), \sum_j \beta_j k(\tilde{\boldsymbol{x}}_j, \boldsymbol{x}) \rangle = \sum_{i,j} \alpha_i \beta_j k(\boldsymbol{x}_i, \tilde{\boldsymbol{x}}_j).$$

Random feature model

Let $\{\phi(\cdot;\omega)\}$ be a collection of random features, π is a prob distribution for of the random variable ω.

$$\mathcal{H}_k = \{f : f(\boldsymbol{x}) = \int a(\omega)\phi(\boldsymbol{x};\omega)d\pi(\omega)\}$$

with $\|f\|^2_{\mathcal{H}_k} = \mathbb{E}_{\omega\sim\pi}[|a(\omega)|^2]$
This is related to the reproducing kernel Hilbert space (RKHS) with kernel:

$$k(\boldsymbol{x},\boldsymbol{x}') = \mathbb{E}_{\omega\sim\pi}[\phi(\boldsymbol{x};\omega)\phi(\boldsymbol{x}';\omega)]$$

Hypothesis space: Given any realization $\{\omega_j\}_{j=1}^m$, i.i.d. with distribution π

$$\mathcal{H}_m(\{\omega_j\}) = \{f_m(\boldsymbol{x},\boldsymbol{a}) = \frac{1}{m}\sum_{j=1}^m a_j\phi(\boldsymbol{x};\omega_j).\}.$$

A priori estimates of the regularized model

$$L_n(\theta) = \hat{\mathcal{R}}_n(\theta) + \lambda\sqrt{\frac{\log(2d)}{n}}\|\theta\|_{\mathcal{H}}, \qquad \hat{\theta}_n = \text{argmin } L_n(\theta)$$

where

$$\|\theta\|_{\mathcal{H}} = \left(\frac{1}{m}\sum_{j=1}^{m}|a_j|^2\right)^{1/2}$$

Theorem

Assume that the target function $f^* : [0,1]^d \mapsto [0,1] \in \mathcal{H}_k$. There exist constants C_0, C_1, C_2, such that for any $\delta > 0$, if $\lambda \geq C_0$, then with probability at least $1-\delta$ over the choice of training set, we have

$$\mathcal{R}(\hat{\theta}_n) \leq C_1\left(\frac{\|f^*\|_{\mathcal{H}_k}^2}{m} + \|f^*\|_{\mathcal{H}_k}\sqrt{\frac{\log(2d)}{n}}\right) + C_2\sqrt{\frac{\log(4C_2/\delta)+\log(n)}{n}}.$$

Outline

Barron spaces

Two-layer neural networks:

$$\frac{1}{m}\sum_{j=1}^{m} a_j \sigma(\boldsymbol{b}_j^T \boldsymbol{x} + c_j)$$

Consider the function $f : D_0 = [0,1]^d \mapsto \mathbb{R}$ of the following form

$$f(\boldsymbol{x}) = \int_\Omega a\sigma(\boldsymbol{b}^T \boldsymbol{x} + c)\rho(da, d\boldsymbol{b}, dc), \quad \boldsymbol{x} \in D_0$$

$\Omega = \mathbb{R}^1 \times \mathbb{R}^d \times \mathbb{R}^1$, ρ is a probability distribution on Ω.
Fourier analog: $\rho(da, d\omega) = \delta(a - \hat{f}(\omega))dad\omega$ (not normalizable).

$$f(\boldsymbol{x}) = \int_{\mathbb{R}^d} \hat{f}(\omega)\cos(\omega^T \boldsymbol{x})d\omega = \int_{\mathbb{R}^1 \times \mathbb{R}^d} a\cos(\omega^T \boldsymbol{x})\rho(da, d\omega)$$

$$\|f\|_{\mathcal{B}_p} = \inf_\rho \left(\mathbb{E}_\rho[|a|^p(\|\boldsymbol{b}\|_1 + |c|)^p]\right)^{1/p}$$

$$\mathcal{B}_p = \{f \in C^0 : \|f\|_{\mathcal{B}_p} < \infty\}$$

What kind of functions admit such a representation?

Theorem (Barron and Klusowski (2016)): If $\int_{\mathbb{R}^d} \|\omega\|_1^2 |\hat{f}(\omega)| d\omega < \infty$, where $\hat{f}$ is the Fourier transform of f, then f can be represented as

$$\tilde{f}(\boldsymbol{x}) = f(\boldsymbol{x}) - (f(0) + \boldsymbol{x} \cdot \nabla f(0)) = \int_{\Omega} a\sigma(\boldsymbol{b}^T \boldsymbol{x} + c)\rho(da, d\boldsymbol{b}, dc)$$

where $\sigma(x) = \max(0, x)$. Moreover $f \in \mathcal{B}_\infty$. Furthermore, we have

$$\|\tilde{f}\|_{\mathcal{B}_\infty} \leq 2 \int_{\mathbb{R}^d} \|\omega\|_1^2 |\hat{f}(\omega)| d\omega$$

Why is this a good idea?

Approximation by two layer networks becomes Monte Carlo integration:

$$f(\boldsymbol{x}) \sim f_m(\boldsymbol{x}) = \frac{1}{m} \sum_{j=1}^{m} a_j \sigma(\boldsymbol{b}_j^T \boldsymbol{x} + c_j)$$

Theorem (Direct Approximation Theorem)

There exists an absolute constant C_0 such that

$$\|f - f_m\|_{L^2(D_0)} \leq \frac{C_0 \|f\|_{\mathcal{B}_2}}{\sqrt{m}}$$

Theorem (Inverse Approximation Theorem)

For $p > 1$, let

$$\mathcal{N}_{p,C} \overset{def}{=} \{ \frac{1}{m} \sum_{k=1}^{m} a_k \sigma(b_k^T \boldsymbol{x} + c_k) : \frac{1}{m} \sum_{k=1}^{m} |a_k|^p (\|b_k\|_1 + c_k)^p \leq C, m \in \mathbb{N}^+ \}.$$

Let f^* be a continuous function. Assume there exists a constant C and a sequence of functions $f_m \in \mathcal{N}_{p,C}$ such that

$$f_m(\boldsymbol{x}) \to f^*(\boldsymbol{x})$$

for all $\boldsymbol{x} \in D_0$, then there exists a probability distribution ρ on Ω, such that

$$f^*(\boldsymbol{x}) = \int a\sigma(\boldsymbol{b}^T \boldsymbol{x} + c)\rho(da, d\boldsymbol{b}, dc),$$

for all $\boldsymbol{x} \in D_0$.

Complexity estimates

Theorem

Let $\mathcal{F}_Q = \{f \in \mathcal{B}_1, \|f\|_{\mathcal{B}_1} \leq Q\}$. *Then we have*

$$\hat{\mathcal{R}}_n(\mathcal{F}_Q) \leq 2Q\sqrt{\frac{2\ln(2d)}{n}}$$

Barron space and RKHS

Equivalent formulation (taking conditional expectation with respect to $\mathbf{w} = (\boldsymbol{b}, c)$):

$$f^*(\boldsymbol{x}) = \int a(\mathbf{w})\sigma(\mathbf{w}^T\boldsymbol{x})\pi(d\mathbf{w}), \quad \boldsymbol{x} = (\boldsymbol{x}, 1)$$

Define:

$$k_\pi(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}_{w\sim\pi}\sigma(\mathbf{w}^T\boldsymbol{x})\sigma(\mathbf{w}^T\boldsymbol{x}')$$

We can write

$$\mathcal{B}_2 = \bigcup_\pi \mathcal{H}_{k_\pi}$$

Shallow neural network can be understood as kernel method with adaptive (learned) kernel.

The ability to learn the right kernel is VERY important.
For example, SVM would be perfect if the right kernel was known.

A priori estimates for regularized model

$$L_n(\theta) = \hat{\mathcal{R}}_n(\theta) + \lambda\sqrt{\frac{\log(2d)}{n}}\|\theta\|_{\mathcal{P}}, \qquad \hat{\theta}_n = \text{argmin } L_n(\theta)$$

where the path norm is defined by:

$$\|\theta\|_{\mathcal{P}} = \frac{1}{m}\sum_{k=1}^{m} |a_k|(\|\boldsymbol{b}_k\|_1 + |c_k|) \quad (= \|f(\cdot;\theta)\|_{\mathcal{B}_1})$$

Theorem (Weinan E, Chao Ma, Lei Wu, submitted)

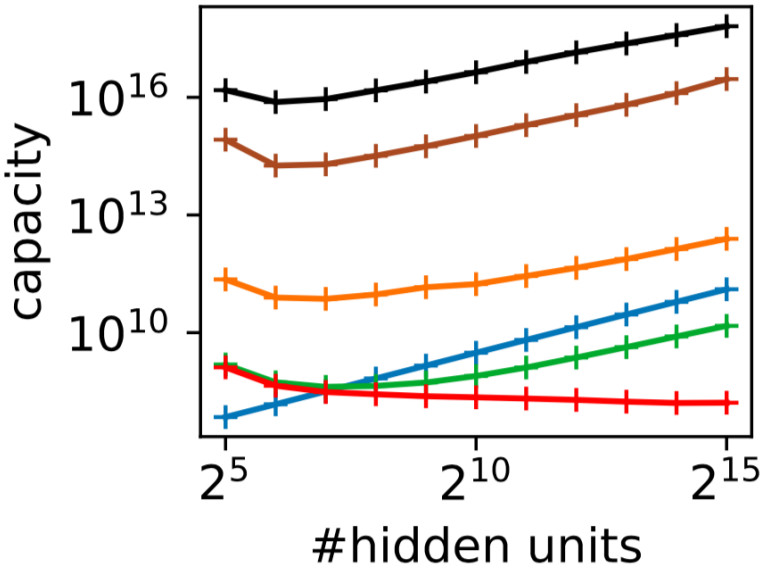
Assume that the target function $f^* : [0,1]^d \mapsto [0,1] \in \mathcal{B}_2$. There exist constants C_0, C_1, C_2, such that for any $\delta > 0$, if $\lambda \geq C_0$, then with probability at least $1-\delta$ over the choice of training set, we have

$$\mathcal{R}(\hat{\theta}_n) \leq C_1\left(\frac{\|f^*\|_{\mathcal{B}_2}^2}{m} + \|f^*\|_{\mathcal{B}_2}\sqrt{\frac{\log(2d)}{n}}\right) + C_2\sqrt{\frac{\log(4C_2/\delta) + \log(n)}{n}}.$$

Traditional results: A posteriori estimates

$$|\mathcal{R}(\theta) - \hat{\mathcal{R}}_n(\theta)| \leq C_1(|||\theta|||+1)\sqrt{\frac{\log(2d)}{n}} + C_2\sqrt{\frac{\log(4C_2(1+|||\theta|||))^2/\delta)}{n}}$$

where $|||\theta|||$ is some norm of θ (see e.g. Behnam Neyshabur, Zhiyuan Li, et al. *Towards Understanding the Role of Over-Parametrization in Generalization of Neural Networks* (2018)).



Outline

Compositional law of large numbers

Consider the following compositional scheme:

$$\begin{aligned} \boldsymbol{z}_{0,L}(\boldsymbol{x}) &= \boldsymbol{V}\boldsymbol{x}, \\ \boldsymbol{z}_{l+1,L}(\boldsymbol{x}) &= z_{l,L}(\boldsymbol{x}) + \frac{1}{L}\boldsymbol{U}_l\sigma \circ (\boldsymbol{W}_l z_{l,L}(\boldsymbol{x})), \end{aligned}$$

$(\boldsymbol{U}_l, \boldsymbol{W}_l)$ are i.i.d. sampled from a distribution ρ.

Theorem

Assume that

$$\mathbb{E}_\rho \| |\boldsymbol{U}||\boldsymbol{W}| \|_F^2 < \infty$$

where for a matrix $\boldsymbol{A}$, $|\boldsymbol{A}|$ means taking element-wise absolute value for $\boldsymbol{A}$. Define $\boldsymbol{z}(\boldsymbol{x}, t)$ by

$$\begin{aligned} \boldsymbol{z}(\boldsymbol{x}, 0) &= \boldsymbol{V}\boldsymbol{x}, \\ \frac{d}{dt}\boldsymbol{z}(\boldsymbol{x}, t) &= \mathbb{E}_{(\boldsymbol{U},\boldsymbol{W})\sim\rho} U\sigma \circ (\boldsymbol{W}\boldsymbol{z}(\boldsymbol{x}, t)). \end{aligned}$$

Then we have

$$\boldsymbol{z}_{L,L}(\boldsymbol{x}) \to \boldsymbol{z}(\boldsymbol{x}, 1)$$

almost surely as $L \to +\infty$.

The compositional function space

$$f_{\alpha,\rho,\boldsymbol{V}}(\boldsymbol{x}) = \alpha^T \boldsymbol{z}(\boldsymbol{x}, 1), \quad \alpha \in \mathcal{R}^D$$

Define the compositional 1-norm and 2-norm as

$$\|f\|_{\mathcal{D}_1} = \inf_{f=f_{\alpha,\rho,\boldsymbol{V}}} \|\alpha\|_1 \left\| e^{\mathbb{E}|\boldsymbol{U}||\boldsymbol{W}|} \right\|_{1,1} \|\boldsymbol{V}\|_{1,1},$$

$$\|f\|_{\mathcal{D}_2} = \inf_{f=f_{\alpha,\rho,\boldsymbol{V}}} \|\alpha\|_F \left\| e^{\sqrt{\mathbb{E}(|\boldsymbol{U}||\boldsymbol{W}|)^2}} \right\|_F \|\boldsymbol{V}\|_F,$$

where $|\cdot|$, $(\cdot)^2$ and $\sqrt{\cdot}$ are element-wise operations. Let

$$\mathcal{D}_i = \{f = f_{\alpha,\rho,\boldsymbol{V}}, \text{ for some } (\alpha, \rho, \boldsymbol{V}), \|f\|_{\mathcal{D}_i} < \infty\}, \quad i = 1, 2,$$

be the compositional function spaces.

Barron space and compositional function space

$$f(\boldsymbol{x}) = \int_{\Omega} a\sigma(\boldsymbol{b}^T\boldsymbol{x} + c)\rho(da, d\boldsymbol{b}, dc)$$

Define $\hat{f}(\boldsymbol{x})$ by

$$\begin{aligned}
\hat{f}(\boldsymbol{x}) &= e_1^T z(\boldsymbol{x}, 1), \\
\frac{d}{dt} z(\boldsymbol{x}, t) &= \mathbb{E}_{\rho(a,\ \boldsymbol{b},\ c)} \begin{bmatrix} a \\ 0 \\ 0 \end{bmatrix} \sigma \circ ([0, \boldsymbol{b}^T, c] z(\boldsymbol{x}, t)), \\
z(\boldsymbol{x}, 0) &= \begin{bmatrix} 0 \\ \boldsymbol{x} \\ 1 \end{bmatrix}.
\end{aligned}$$

Then, $f(\boldsymbol{x}) = \hat{f}(\boldsymbol{x})$. In addition, we have

Theorem

$\mathcal{B}_2 \subset \mathcal{D}_2$. There exists constant $C > 0$, such that

$$\|f\|_{\mathcal{D}_2} \leq \sqrt{d+1}\|f\|_{\mathcal{B}_2}$$

holds for any $f \in \mathcal{B}_2$,

Extension

Let $\{\rho_t\}$ be a family of prob distributions (for $(\boldsymbol{U}, \boldsymbol{W})$) such that $\mathbb{E}_{\rho_t} g(\boldsymbol{U}, \boldsymbol{W})$ is integrable as a function of t for any continuous function g. Define:

$$\begin{aligned} \boldsymbol{z}(\boldsymbol{x}, 0) &= \boldsymbol{V}\boldsymbol{x}, \\ \frac{d}{dt}\boldsymbol{z}(\boldsymbol{x}, t) &= \mathbb{E}_{(\boldsymbol{U}, \boldsymbol{W}) \sim \rho_t} \boldsymbol{U}\sigma \circ (\boldsymbol{W}\boldsymbol{z}(\boldsymbol{x}, t)) \end{aligned}$$

We can also define compositional space for this case, e.g. we define $f_{\alpha, \{\rho_t\}, \boldsymbol{V}}(\boldsymbol{x}) = \alpha^T \boldsymbol{z}(\boldsymbol{x}, 1)$ and

$$\begin{aligned} \frac{d}{dt}\mathbf{N}(t) &= \mathbb{E}_{\rho_t} |\boldsymbol{U}||\boldsymbol{W}| \mathbf{N}(t), \\ \mathbf{N}(0) &= \mathbf{I} \\ \|f\|_{\mathcal{D}_1} &= \inf_{f = f_{\alpha, \{\rho_t\}, \boldsymbol{V}}} \|\alpha\|_1 \left\|\mathbf{N}(1)\right\|_{1,1} \|\boldsymbol{V}\|_{1,1}, \end{aligned}$$

$\|\cdot\|_{\mathcal{D}_2}$ is defined similarly.

Inverse approximation theorem

Let $f \in L^2(D_0)$. Assume that there is a sequence of residual networks $\{f_L(\boldsymbol{x})\}_{L=1}^{\infty}$ with increasing depth such that $\|f(\boldsymbol{x}) - f_L(\boldsymbol{x})\| \to 0$. Assume further that the parameters are (entry-wise) bounded, then there exists α, $\{\rho_t\}$ and $\boldsymbol{V}$ such that

$$f(\boldsymbol{x}) = f_{\alpha,\{\rho_t\},\boldsymbol{V}}(\boldsymbol{x}).$$

Theorem (Direct approximation theorem)

Let $f \in L^2(D_0) \cap \mathcal{D}_2$. There exists a residue-type neural network $f_L(\cdot;\tilde{\theta})$ of input dimension $d+1$ and depth L such that $\|f_L\|_P \lesssim \|f\|_{c_1}^3$ and

$$\int_{D_0} |f(\boldsymbol{x}) - f_L((\boldsymbol{x},1);\tilde{\theta})|^2 dx \to 0 \lesssim \frac{\|f\|_{c_2}^2}{L}$$

Furthermore, if $f = f_{\alpha,\{\rho_t\},\boldsymbol{V}}$ and ρ_t is Lipschitz continuous in t, then

$$\int_{D_0} |f(\boldsymbol{x}) - f_L((\boldsymbol{x},1);\tilde{\theta})|^2 dx \lesssim \frac{\|f\|_{\mathcal{D}_2}^2}{L}$$

Complexity control

Rademacher complexity bound for path norm

Let $\mathcal{F}_{L,Q} = \{f_L : \|f_L\|_{\mathcal{D}_1} \leq Q\}$. Assume $\boldsymbol{x}_i \in [-1,1]^d$. Then, for any data set $S = (\boldsymbol{x}_1, ..., \boldsymbol{x}_n)$, we have

$$\hat{R}_S(\mathcal{F}_{L,Q}) \leq 3Q\sqrt{\frac{2\log(2d)}{n}}.$$

Theorem (A posteriori estimates)

Assume that the target function $\|f^\star\|_\infty \leq 1$. For the target function $f^\star$ and residue-type network $f_L(\boldsymbol{x};\theta)$ given above, with probability at least $1-\delta$,

$$|L(\theta) - L_n(\theta)| \leq 24(\|\theta\|_{\mathcal{D}_1} + 1)\sqrt{\frac{2\log(2d)}{n}} + 4\sqrt{\frac{2\log[4(\|\theta\|_{\mathcal{D}_1} + 1)^2/\delta]}{n}}.$$

Regularized model and a priori estimates

Regularized loss function:

$$J(\theta) = \hat{L}(\theta) + \lambda(\|\theta\|_{\mathcal{D}_1} + 1)\sqrt{\frac{2\log(2d)}{n}}.$$

Theorem (A-priori estimate)

Assume that $f^\star : [-1,1]^d \to [-1,1]$ *such that* $f^* \in \mathcal{D}_2$. *Let*

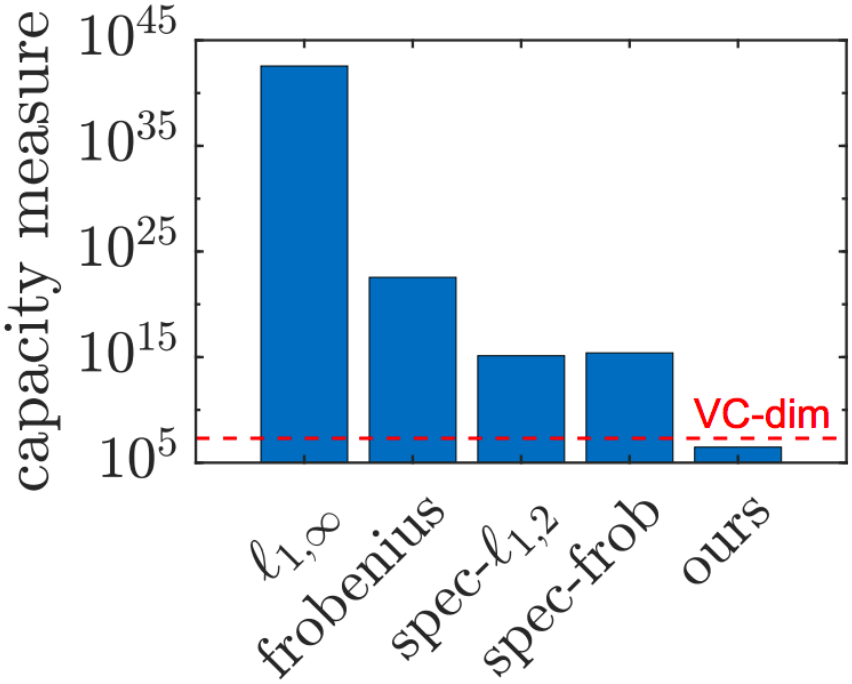
$$\hat{\theta} = \textit{argmin}_{\,\theta} J(\theta)$$

then if λ *is larger than some constant, and the depth* L *is sufficiently large, for any* $\delta > 0$, *with probability at least* $1 - \delta$,

$$L(\hat{\theta}) \lesssim \frac{\|f^*\|^2_{\mathcal{D}_2}}{L} + \lambda(\|f^*\|^3_{\mathcal{D}_1} + 1)\sqrt{\frac{\log(2d)}{n}} + \sqrt{\frac{\log(1/\delta)}{n}}.$$

Comparison with other results

Fully-connected deep networks $f_L(\boldsymbol{x};\theta) = \boldsymbol{W}_L\sigma(\boldsymbol{W}_{L-1}\sigma(\cdots\sigma(\boldsymbol{W}_1\boldsymbol{x})))$



From Arora, Ge, Neyshabur and Zhang (2018).
Neyshabur et al (2015), Yao (2015): exponential factors in the complexity estimates.

Comparison with other results

Barron et al. (2018)

$$|||\theta||| = \frac{1}{L}\sqrt{V}\sum_{l=1}^{L}\sum_{j_l}\sqrt{V_{j_l}^{\text{in}}V_{j_l}^{\text{out}}},$$

where $V = |||\boldsymbol{W}_L|\cdots|\boldsymbol{W}_1|||_1$, $V_{j_l}^{\text{in}} = \left|\left||\boldsymbol{W}_{l,[j_l,:]}||\boldsymbol{W}_{l-1}|\cdots|\boldsymbol{W}_1|\right|\right|_1$ and $V_{j_l}^{\text{out}} = \left|\left||\boldsymbol{W}_L|\cdots|\boldsymbol{W}_{l+1}||\boldsymbol{W}_{l,[:,l_j]}|\right|\right|_1$.

$$\hat{R}_S(f_L(\boldsymbol{x};\theta) : |||\theta||| \leq Q) \leq CQL\sqrt{(L-2)\log D + \log(8ed)}\frac{\log n}{\sqrt{n}}$$

Comparison with other results

Bartlett et al. (2017) proposed the spectral complexity norm

$$|||\theta||| = \left[\prod_{l=1}^{L} \|\boldsymbol{W}_l\|_\sigma\right] \left[\sum_{l=1}^{L} \frac{\|\boldsymbol{W}_l^\mathsf{T}\|_{2,1}^{2/3}}{\|\boldsymbol{W}_l\|_\sigma^{2/3}}\right]^{3/2},$$

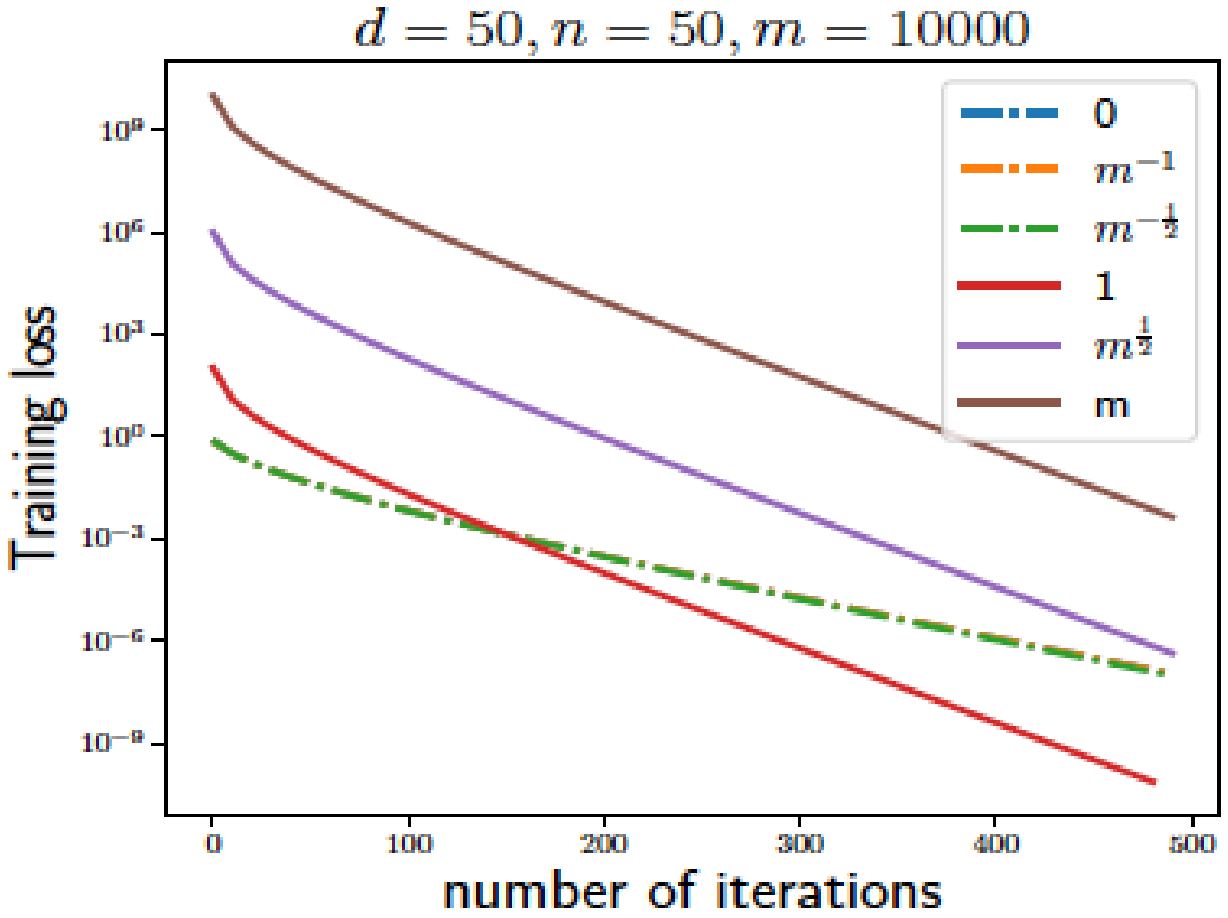
where $\|\cdot\|_\sigma$ denotes the spectral norm and $\|\cdot\|_{p,q}$ denotes the (p,q) matrix norm $\|\boldsymbol{W}\|_{p,q} = \|(\|\boldsymbol{W}_{:,1}\|_p, \cdots, \|\boldsymbol{W}_{:,m}\|_p)\|_q$.

$$\hat{R}_S(f_L(x;\theta) : |||\theta||| \leq Q) \leq \frac{12\log n}{\sqrt{n}}\sqrt{2\log(2D)}Q.$$

$$L(\hat{\theta}) \leq \frac{16\gamma^2(f^\star)}{D'L} + 6\sqrt{\frac{1}{2n}\log\frac{7}{\delta}}$$

$$+\frac{1}{\sqrt{n}}\left[12(\lambda+4)\log n\sqrt{2\log(2D)}+3\right]\left[L^{3/2}ed\gamma(f^\star)+1\right].$$

Outline


$d = 50, n = 50, m = 10000$
Training loss
number of iterations
0
m^{-1}
$m^{-\frac{1}{2}}$
1
$m^{\frac{1}{2}}$
m
10^{9}
10^{6}
10^{3}
10^{0}
10^{-3}
10^{-6}
10^{-9}
0
100
200
300
400
500

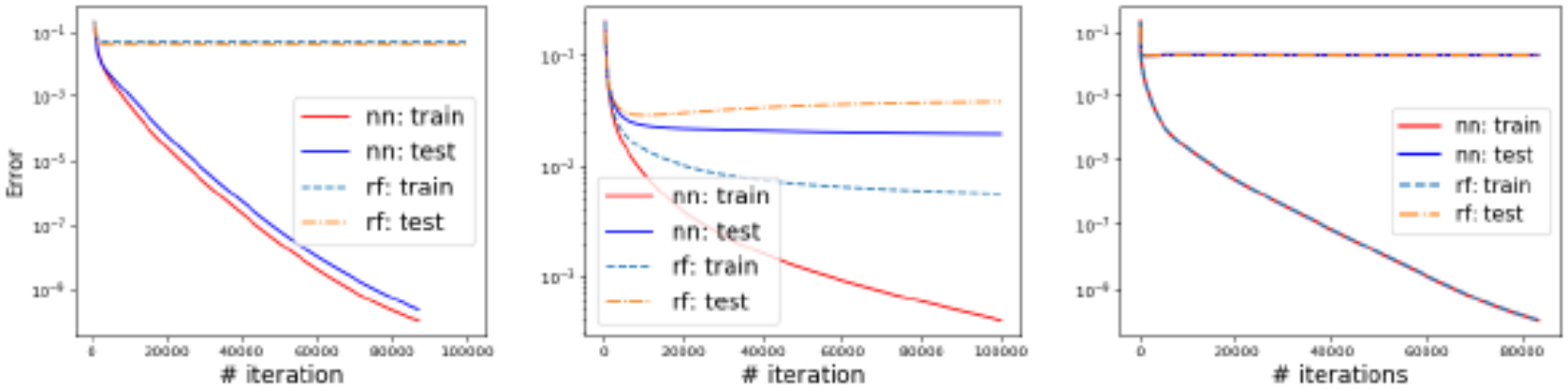


Figure 2: Training and testing errors of neural network and random feature models during training, starting from zero initialization of a. Left: m=4; Middle: m=50; Right: m=1000.

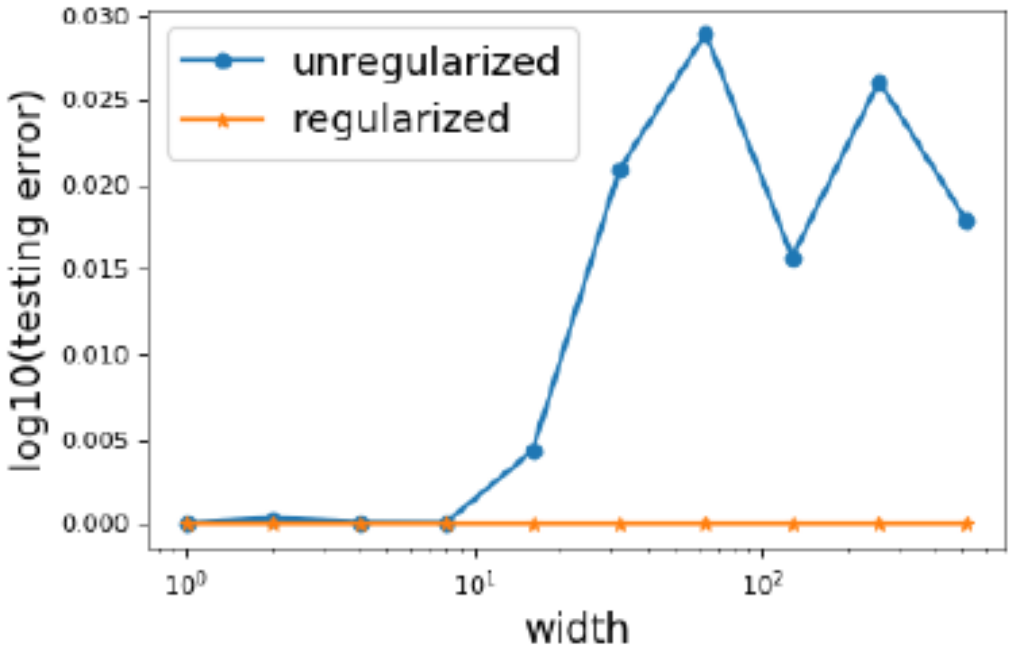


Figure 3: Testing errors of two-layer neural networks for different width, compared with the regularized neural network using path norm. In the regularized problem, we choose $\lambda = 0.01$.

Gradient descent dynamics for un-regularized model

Over-parametrized regime $m \geq O(n^\alpha)$: CLT scaling

$$f_0(\boldsymbol{x}) = \frac{1}{\sqrt{m}} \sum_{k=1}^{m} a_k^0 \sigma(\boldsymbol{x}^T \boldsymbol{b}_k^0)$$

- Optimization: Empirical risk goes to 0?
  - S. Du et al. (2018): With random (normal) initialization, GD can fit any target (random labels).
- Generalization: Population risk (real error) small? ("**Implicit regularization**")
  - early stopping: Z. Allen-Zhu et al. (2018), $a_k^0 = \pm\varepsilon$ then fixed in time.
  - long time analysis: Sanjeev Arora et al. (2019), $a_k^0 = \pm 1$ and then fixed in time, $b_k^0 = N(0, \kappa^2 I_d)$.

  Error estimates involve ε or κ.

Related work:

- A. Daniely (2017), Z. Allen-Zhu et al. (2018), Y. Cao et al. (2019).

Gradient descent dynamics for un-regularized model

$$f_0(\boldsymbol{x}) = \sum_{k=1}^{m} a_k^0 \sigma(\boldsymbol{x}^T \boldsymbol{b}_k^0)$$

$a_k = \pm\beta = o(1), \boldsymbol{b}_k$ i. i. d. from uniform distribution on $\mathbb{S}^{d-1}$.

- optimization: exponential convergence of the empirical risk for very general initialization
- generalization: **no better than the random feature model**

Gradient descent for two-layer networks (un-regularized)

Case 1: Over-parametrized regime $m \geq O(n^{\alpha})$:

GD solutions for two-layer neural networks and random feature based kernel method with the kernel given by

$$k_0(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}_{\mathbf{w}\sim\rho_0}\sigma(\mathbf{w}^T\boldsymbol{x})\sigma(\mathbf{w}^T\boldsymbol{x}')$$

are uniformly close.

Theorem

For any $\delta > 0$, as long as $m > 64\lambda_n^{-4}\ln(n^2/\delta)$, with probability at least $1-\delta$ we have,

$$\|f_m(\boldsymbol{x};\Theta_t) - f_m(\boldsymbol{x};\Theta_0) - f^{ker}(\boldsymbol{x}, mt)\| \leq \frac{c(\delta)n\ln^{\frac{3}{2}}(n)}{\lambda_n^2}\left(\frac{1}{\sqrt{m}} + \beta + \sqrt{m}\beta^3\right),$$

where $c(\delta) \leq C_1(1+\sqrt{\ln(1/\delta)})^2$ for some absolute constant C_1.

The implicit regularization effect in shallow networks is NO better than that of the kernel method in the over-parametrized regime.

Gradient descent for two-layer networks (un-regularized)

Case 2: Arbitrary values of m, n

Theorem

Assume that $f^* \in \mathcal{H}_{k_0}$ and $\|f^*\|_\infty \leq 1$. Assume that the norm of the output layer $\|a\|_F$ at initialization is $\mathcal{O}(\frac{1}{\sqrt{m}})$. Then "with high probability" there exists constant C, s.t.

$$\|f_m(\boldsymbol{x};\Theta_s) - f^*(\boldsymbol{x})\|_2^2 \leq C\left(\frac{1}{m}+\frac{1}{mt}+\frac{1}{\sqrt{n}}\left(1+\sqrt{t}+\frac{\sqrt{mt}}{n^{1/4}}\right)^2 + \frac{t^2}{m^2}(1+mt)^2\left(1+\frac{t^2}{m^2}(t+m)^4\right)\left(1+\sqrt{t}+\frac{\sqrt{mt}}{n^{1/4}}\right)^2\right). \quad (1)$$

1. early stopping: if $m > n$ and take $t = \mathcal{O}(\frac{\sqrt{n}}{m})$, then

$$\|f_m(\boldsymbol{x};\Theta_s) - f^*(\boldsymbol{x})\|_2^2 \leq \mathcal{O}(\frac{1}{m}) + \mathcal{O}(\frac{1}{\sqrt{n}}).$$

2. limited to target functions in the RKHS, still related to kernel method.

Curse of dimensionality

Current understanding about "implicit regularization":
Implicit regularization might hold but we need to know which RKHS space that the target function lives in.

Theorem

For any fixed training data $S = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n\}$ and target function f^*, denote the two-layer neural network of width m found by GD with initialization Θ_0 by $f_{m,\infty}(\cdot;\ f^*, S, \Theta_0)$ Then,

$$\sup_{\|f^*\|_{\mathcal{B}_2} \leq 1} \|f - f_{m,\infty}(\cdot;\ f^*, S, \Theta_0)\|_{L^2(D_0)} \geq \frac{\kappa}{d} \left(\frac{1}{n+1}\right)^{1/d},$$

where κ is a constant independent of d, and n.

Training deep neural networks

Consider deep neural networks with skip-connections:

$$\begin{aligned}\boldsymbol{h}^{(0)} &= (\boldsymbol{x}^T, 0)^T \in \mathbb{R}^{d+1} \\ \boldsymbol{h}^{(l+1)} &= \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{h}^{(l)}_{d+1} \end{pmatrix} + \boldsymbol{U}^{(l)} \sigma \circ (\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l)}) \\ f(\boldsymbol{x}; \Theta) &= \boldsymbol{V}^T \boldsymbol{h}^{(L)},\end{aligned}$$

where $\boldsymbol{U}^{(l)} \in \mathbb{R}^{(d+1)\times m}, \boldsymbol{W}^{(l)} \in \mathbb{R}^{m\times(d+1)}$. Let $\Theta = \{\boldsymbol{U}^{(l)}, \boldsymbol{W}^{(l)}\}_{l=1}^{L}$ denote all the trainable parameters ($\mathbf{w}$ is fixed).

Initialization

$$\boldsymbol{U}_0^{(l)} = 0 \qquad \boldsymbol{W}_0^{(l)} = \begin{pmatrix} \boldsymbol{B}_0^{(l)} & 0 \end{pmatrix},$$

where $\boldsymbol{B}_0^{(l)} \in \mathbb{R}^{m\times d}$ with each row sampled from $\mathcal{N}(0, I_d/m)$.

Training deep neural networks (Cont'd)

Theorem (Optimization)

Let $K \in \mathbb{R}^{n\times n}$ *with* $K_{i,j} = \frac{1}{n}\mathbb{E}_{w\sim\pi_0}[\sigma(w^T\boldsymbol{x}_i)\sigma(w^T\boldsymbol{x}_j)]$, *and we assume that* $\lambda_n = \lambda_{\min}(K) > 0$. *Then there exist a constant* C *such that for* $L \geq \frac{C}{\lambda_n^3}\ln^2(n/\delta)$, *we have with probability* $1-\delta$ *over the initialization* Θ_0, *the following inequality holds for any* $t \geq 0$,

$$\hat{\mathcal{R}}_n(\Theta_t) \leq e^{-\frac{\lambda_n L}{2}}\hat{\mathcal{R}}_n(\Theta_0).$$

All recent works require that network width $\geq \mathbf{poly}(L, n)$, this result only requires the width to be larger than $d+1$.

Training deep neural networks (Cont'd)

Theorem (Generalization)

Assume $f^*(\boldsymbol{x}) = \int a^*(\omega)\sigma(\omega^T\boldsymbol{x})d\pi_0(\omega)$. π_0 *is the uniform distribution over sphere* $\mathbb{S}^{d-1}$. *Assume that there exists constants* C_1, C_2 *such that* $|a^*(\omega)| \leq C_1$, *and* $L \geq C_2 n^{1/4}$. *Then for any* $\delta \in (0,1)$, *with probability* $1-\delta$ *over the choice of the random initialization, we have for* $t \leq 1$

$$\mathcal{R}(\Theta_t) \lesssim \frac{\|f^*\|_{\mathcal{H}_k}^2}{mL} + \frac{\|f^*\|_{\mathcal{H}_k}}{\sqrt{n}} + \frac{tL}{\sqrt{n}} + \frac{\|f^*\|_{\mathcal{H}_k}^2}{tL} + \varepsilon(m, L, n, t, \delta),$$

where

$$\varepsilon(m, L, n, t, \delta) = \frac{C_1^2 \log(1/\delta)}{mL} + \sqrt{\frac{\log(Lt)}{n}} + \frac{C_1^2 \log(1/\delta)}{\sqrt{mLt}}$$

Outline

Concluding remarks

Analogy with Monte Carlo

$$(I(g) - I_n(g))^2 \sim \frac{\gamma(g)}{n}$$

$\gamma(g) =$ some kind of variance of g, depending on the details of the Monte Carlo algorithm.

For the ML models considered

$$\mathcal{R}(\theta) \leq \frac{\gamma_1(f^*)}{m} + \frac{\gamma_2(f^*)}{\sqrt{n}}$$

For this purpose, we need to:

- find probablistic interpretation of the machine learning model
- identify the right function spaces/norms for approximation theory
- study the Rademacher complexity

- Should think in terms of approximating probability distributions (ρ, $\{\rho_t\}$), not specific parameters or weights.
- Regularization is important for obtaining well-posed models. Relying on "implicit regularization" might NOT be the way to go.

A main mathematical question is to identify and study **low complexity spaces** in high dimension.

Open problems

- What about these function spaces? how big are they? size of the norms
- classification problems
- other regularizations?
- other network structures (e.g. DenseNet)?
- other activation functions?

A priori estimates for regularized kernel method

Theorem

For any fixed $\lambda > 0$,
Assume $f^* \in \mathcal{H}_k$, *and let* $\hat{f}_n$ *be the solution of the ridge regression given by*

$$\hat{f}_n \overset{def}{=} \textit{argmin}_{\, f \in \mathcal{H}_k} \frac{1}{n} \sum_{i=1}^{n} (f(\boldsymbol{x}_i) - f^*(\boldsymbol{x}_i))^2 + \frac{\lambda}{\sqrt{n}} \| f \|_{\mathcal{H}_k}.$$

We have that

$$\mathbb{E}_{\boldsymbol{x}}[|\hat{f}_n(\boldsymbol{x}) - f^*(\boldsymbol{x})|^2] \leq C(\lambda) \frac{\|f^*\|_{\mathcal{H}_k}}{\sqrt{n}}$$

Random feature model and the kernel method

Let $\{\phi(\cdot;\omega)\}$ be a collection of random features, π is a prob distribution for of the random variable ω.

$$k(\boldsymbol{x},\boldsymbol{x}') = \mathbb{E}_{\omega\sim\pi}[\phi(\boldsymbol{x};\omega)\phi(\boldsymbol{x}';\omega)]$$

We can express functions in $\mathcal{H}_k$ by

$$f(\boldsymbol{x}) = \int a(\omega)\phi(\boldsymbol{x};\omega)d\pi(\omega)$$

with $\|f\|^2_{\mathcal{H}_k} = \mathbb{E}_{\omega\sim\pi}[|a(\omega)|^2]$

Hypothesis space: Given any realization $\{\omega_j\}_{j=1}^m$, i.i.d. with distribution π

$$\mathcal{H}_m(\{\omega_j\}) = \{f_m(\boldsymbol{x},\boldsymbol{a}) = \frac{1}{m}\sum_{j=1}^m a_j\phi(\boldsymbol{x};\omega_j).\}.$$

"Implicit regularization" of gradient descent

$$\frac{da_j}{dt} = -\nabla_{a_j}\hat{\mathcal{R}}_n(\boldsymbol{a}), \quad \hat{\mathcal{R}}_n(\boldsymbol{a}) = \frac{1}{n}\sum_{i=1}^{n}(f_m(\boldsymbol{x}_i, \boldsymbol{a}) - f^*(\boldsymbol{x}_i))^2$$

Case 1: Arbitrary values of m, n: With the special initialization

$$a_j(0) = 0$$

Theorem

Let $f^*(\boldsymbol{x}) = \int a^*(\omega)\phi(\boldsymbol{x};\omega)d\pi(\omega)$ *with* $|a^*(\omega)| \leq C$. *Let* $\boldsymbol{a}(\cdot)$ *be the solution defined above. For any* $\delta \in (0,1)$ *then with probability* $1-\delta$ *over the choice of* $\{\omega_j\}$ *we have* $(k = k_\pi)$

$$\mathbb{E}_{\boldsymbol{x}}[(f_m(\boldsymbol{x};\boldsymbol{a}(t)) - f^*(\boldsymbol{x}))^2] \lesssim \underbrace{\frac{\|f^*\|_{\mathcal{H}_k}^2}{m}}_{\text{Approx Err.}} + \underbrace{\frac{t}{\sqrt{n}}}_{\text{Estimation Err.}} + \underbrace{\frac{\|f^*\|_{\mathcal{H}_k}^2}{2t}}_{\text{Optimization Err.}} + \varepsilon(m,n,t,\delta),$$

where

$$\varepsilon(m,n,t,\delta) = \frac{C^2\log(1/\delta)}{m} + \sqrt{\frac{\log(t)}{n}} + \frac{C\log(1/\delta)}{\sqrt{mt}}$$

Implicit regularization

Case 2: Large m, large t. General initialization

Let $\Phi \in \mathbb{R}^{m\times n}$ with $\Phi_{k,i} = \phi(\boldsymbol{x}_i; \omega_k)$ denote the feature map.

As $t \to \infty$, GD solution converges to

$$\boldsymbol{a}_\infty = \underbrace{(1 - P_\Phi)\boldsymbol{a}_0}_{(1)} + \underbrace{m\Phi(\Phi^T\Phi)^{-1}\boldsymbol{y}}_{(2)}$$

- (1): P_Φ is the orthogonal projection to the column range of Φ.
- (2): This part is the minimum norm solution. Denote it by $\hat{\boldsymbol{a}}_{n,m}$.

Theorem (Generalization ability of the minimum norm solution)

Let $f^*(\boldsymbol{x}) = \int a^*(\omega)\phi(\boldsymbol{x};\omega)d\pi(\omega)$ with $|a^*(\omega)| \le C$. Denote the kernel matrix by $K \in \mathbb{R}^{n\times n}$ with $K_{i,j} = \frac{1}{n}k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. If $m \ge \lambda_{min}^{-2}(K)$, then for any $\delta \in (0,1)$ with probability $1-\delta$ over the choice $\{\omega_j\}$, we have

$$\mathbb{E}_{\boldsymbol{x}}[|f_m(\boldsymbol{x}; \hat{\boldsymbol{a}}_{n,m}) - f^*(\boldsymbol{x})|^2] \lesssim \frac{\|f^*\|_{\mathcal{H}_k}^2}{m} + \frac{\|f^*\|_{\mathcal{H}_k}}{\sqrt{n}} + C^2\frac{\log(1/\delta)}{m} + C\sqrt{\frac{\log(1/\delta)}{n}}.$$