PAC Bound of Neural Networks

Dawei Sun

daweis2@illinois.edu
University of Illinois at Urbana Champaign

From ealier lectures, we already know that the performance of a hypothesis class $\mathcal{F}$ can be bounded in terms of the expected Rademacher average $\mathbb{E}\left[R_n(\mathcal{F}(X^n))\right]$. Next, we will give a bound on $\mathbb{E}\left[R_n(\mathcal{F}(X^n))\right]$ for function class $\mathcal{F}$ induced by neural networks and $X_i \in \mathbb{R}^d$.

First, consider the Rademacher average of the function class induced by the linear component of a single layer neural network, i.e. $\mathcal{F} = \{\langle w, \cdot \rangle : \forall w \in \mathbb{R}^d \text{ and } ||w|| \leq B\}$. Since $\mathcal{F}$ is a d-dimensional Dudley class, a naive bound imediately follows. However, since we have some constraints on functions in $\mathcal{F}$ (linear and bounded weights), we can derive a tighter bound. Rademacher aveage for this class can be bounded as follows.

$$\begin{aligned}
R_n(\mathcal{F}(X^n)) &= \frac{1}{n}\mathbb{E}\left[\sup_{w\in\mathbb{R}^d, ||w||\leq B}\left|\sum_{i=1}^n \epsilon_i \langle w, X_i\rangle\right|\right] \\
&= \frac{1}{n}\mathbb{E}\left[\sup_{w\in\mathbb{R}^d, ||w||\leq B}\left|\langle w, \sum_{i=1}^n \epsilon_i X_i\rangle\right|\right] \\
&= \frac{1}{n}\mathbb{E}\left[B\left|\left|\sum_{i=1}^n \epsilon_i X_i\right|\right|\right] \\
&= \frac{B}{n}\mathbb{E}\left[\sqrt{\sum_{i=1}^n\sum_{j=1}^n \epsilon_i\epsilon_j\langle X_i, X_j\rangle}\right] \\
&\leq \frac{B}{n}\cdot\sqrt{\mathbb{E}\left[\sum_{i=1}^n\sum_{j=1}^n \epsilon_i\epsilon_j\langle X_i, X_j\rangle\right]} \\
&= \frac{B}{n}\cdot\sqrt{\sum_{i=1}^n ||X_i||^2},
\end{aligned}$$

where ϵ_i's the Rademacher random variables. In the third step of the derivation, we use the fact that $w^* = B \cdot \frac{\sum_{i=1}^n \epsilon_i X_i}{||\sum_{i=1}^n \epsilon_i X_i||}$. In the fifth step, we use Jensen's inequality. In the last step, we use the fact that $\mathbb{E}\left[\epsilon_i \epsilon_j\right] = \mathbf{1}_{\{i=j\}}$. Furthermore, if random variables $||X_i|| \leq R$ for all i, then we have

$$\mathbb{E}\left[R_n(\mathcal{F}(X^n))\right] \leq \frac{BR}{\sqrt{n}}. \tag{1}$$

Note that since we use the linearity and bounded-norm constriants of the function class, the derived bound is dimension-free.

Now, consider the above linear model composed with nonlinear operations (activations), i.e. the function class $\mathcal{F} = \{\sigma(\langle w, \cdot \rangle) : \forall w \in \mathbb{R}^d \text{ and } ||w|| \leq B\}$, where $\sigma : \mathbb{R} \mapsto \mathbb{R}$ is the activation function. Suppose σ is Lipschitz-continuous with Lipschitz constant L and $\sigma(0) = 0$. Using the contraction principle, we have

$$\begin{aligned} R_n(\mathcal{F}(X^n)) &= \frac{1}{n}\mathbb{E}\left[\sup_{w \in \mathbb{R}^d, ||w|| \leq B} \left| \sum_{i=1}^n \epsilon_i \sigma(\langle w, X_i \rangle) \right| \right] \\ &\leq \frac{2L}{n}\mathbb{E}\left[\sup_{w \in \mathbb{R}^d, ||w|| \leq B} \left| \sum_{i=1}^n \epsilon_i \langle w, X_i \rangle \right| \right] \\ &\leq \frac{2LB}{n} \cdot \sqrt{\sum_{i=1}^n ||X_i||^2}. \end{aligned}$$

Again, if random variables $||X_i||$ for all i are supported on a ball with radius R, then we have

$$\mathbb{E}\left[R_n(\mathcal{F}(X^n))\right] \leq \frac{2LBR}{\sqrt{n}}. \tag{2}$$

Now, we consider a multi-layer neural network. First, define a function $N_{\sigma,w} : \mathbb{R}^m \mapsto \mathbb{R}$ which charateristic a single neuron with σ as activation and $w \in \mathbb{R}^m$ as weights:

$$N_{\sigma,w}(x_1, \cdots, x_m) = \sigma(\sum_{i=1}^n w_i x_i),$$

and denote $N_{\sigma,w}$ composed with m real-valued functions $h_1, \cdots, h_m$ by

$$N_{\sigma,w} \circ (h_1, \cdots, h_m)(x) = N_{\sigma,w}(h_1(x), \cdots, h_m(x)).$$

Let $\mathcal{G}$ be a family of base classifiers from $\mathbf{X} \subseteq \mathbb{R}^d$ to $\mathbb{R}$. A multilayer neural networks are created by repeating this composition operation. For a ℓ-layer neural network, let $\sigma_1, \cdots, \sigma_\ell$ be a sequence of activation functions for each layer respectively. Suppose that σ_i is Lipschitz continuous with Lipschitz constant L_i and $\sigma_i(0) = 0$ for all $i \in [\ell]$. Let $B_1, \cdots, B_\ell$ be a sequence of positive reals. We can define $\ell + 1$ function classes $\mathcal{F}_0, \cdots, \mathcal{F}_\ell$ recursively as follows:

$$\mathcal{F}_0 = \mathcal{G},$$

and for $1 \leq j \leq \ell$,

$$\mathcal{F}_j = \left\{ N_{\sigma_j, w} \circ (f_1, \cdots, f_m) : \; m \in \mathbb{N}; \; \sum_{i=1}^{n} |w_i| \leq B_j; \; f_k \in \mathcal{F}_{j-1}, \forall k \in [m] \right\}.$$

First, we notice that $\{\sum_{i=1}^m w_i f_i : \; m \in \mathbb{N}; \; \sum_{i=1}^n |w_i| \leq 1; \; f_k \in \mathcal{F}_{j-1}, \forall k \in [m]\}$ is the absoulte convex hull of $\mathcal{F}_{j-1}$. Thus,

$$\mathcal{F}_j = \sigma_j \circ (B_j \cdot \text{absconv}(\mathcal{F}_{j-1})).$$

Consider the last layer, we have $\mathcal{F}_\ell = \sigma_\ell \circ (B_\ell \cdot \text{absconv}(\mathcal{F}_{\ell-1}))$, and thus $\mathcal{F}_\ell(X^n) = \sigma_\ell \circ (B_\ell \cdot \text{absconv}(\mathcal{F}_{\ell-1}(X^n)))$. Therefore,

$$\begin{aligned} R_n(\mathcal{F}_\ell(X^n)) &= R_n(\sigma_\ell \circ (B_\ell \cdot \text{absconv}(\mathcal{F}_{\ell-1}(X^n)))) \\ &\leq 2L_\ell R_n(B_\ell \cdot \text{absconv}(\mathcal{F}_{\ell-1}(X^n))) \\ &= 2L_\ell B_\ell R_n(\mathcal{F}_{\ell-1}(X^n)), \end{aligned}$$

where we use the contration principle in the second step and Rademacher average of convex hull in the last step. By doing this repeatedly, we arrived at the bound for the neural network:

$$R_n(\mathcal{F}_\ell(X^n)) \leq 2^\ell \prod_{i=1}^{\ell} L_i B_i R_n(\mathcal{G}(X^n)). \tag{3}$$

This bound is also dimension-free, i.e. indepent of the width of each layer. It only depends on the Lipschitz constants, the L-1 norm of the weights and the number of layers ℓ. However, due to the use of the contraction principle, the bound is multiplied by 2 after adding a new layer. We denote $\prod_{i=1}^{\ell} L_i B_i$ by M. Then, in order to guarantee the bound not to diverge too fast, we need $M \leq 2^\ell$, which is not reasonable for commonly-used neural networks, e.g. ReLU neural networks.

In a recent paper[1], the authors showed that the factor 2^ℓ can be reduced to $\sqrt{\ell}$. Next, we give a sketch of their derivation. First, we need the following lammas.

Lemma 1. *Let $\mathcal{F}'$ be a real-valued function class on $\mathbf{X} \subseteq \mathbb{R}^d$ and $\mathcal{F}$ be $\sigma \circ (B \cdot absconv(\mathcal{F}'))$, for some Lipschitz continuous function $\sigma : \mathbb{R} \mapsto \mathbb{R}$ with Lipschitz constant L and $\sigma(0) = 0$. Let $G : \mathbb{R} \mapsto \mathbb{R}$ be a convex nondecreasing positive[1] function. Then*

$$\mathbb{E}\left[\sup_{f \in \mathcal{F}} G\left(\left|\sum_{i=1}^n \epsilon_i f(X_i)\right|\right)\right] \leq 2 \cdot \mathbb{E}\left[\sup_{f' \in \mathcal{F}'} G\left(LB \cdot \left|\sum_{i=1}^n \epsilon_i f'(X_i)\right|\right)\right].$$

Proof. (sketch) First, use the fact that $G(|u|) \leq G(u) + G(-u)$ to remove the absolute value sign. Then use a generalization of the contraction principle to involve $\mathcal{F}'$. Finally, use Hölder's inequality to add the absolute value sign back. □

Lemma 2. *Let $\mathcal{A}$ be a bounded subset of $\mathbb{R}^n$. Then, for any $\lambda > 0$,*

$$\mathbb{E}\left[\exp\left(\lambda \sup_{a \in \mathcal{A}} \left|\sum_{i=1}^n \epsilon_i a_i\right|\right)\right] \leq \exp\left(\frac{\lambda^2}{2} \sum_{i=1}^n \sup_{a \in \mathcal{A}} |a_i|^2\right) \exp\left(\lambda n R_n(\mathcal{A})\right).$$

Proof. (sketch) The proof uses the fact that $\sup\limits_{a \in \mathcal{A}} |\sum_{i=1}^n \epsilon_i a_i|$ is a deterministic function of ϵ^n and has bounded difference. Then, mimicking the proof of McDiarmid's inequality gives the result. □

With these two lemmas, we can derive the following theorem.

Theorem 1. *For any $X_1, \cdots, X_n$,*

$$R_n(\mathcal{F}_\ell(X^n)) \leq M \cdot \left(R_n(\mathcal{G}(X^n)) + \frac{2}{n}\sqrt{\ell \log 2 \cdot \sum_{i=1}^n \sup_{g \in \mathcal{G}} |g(X_i)|^2}\right).$$

[1] We added the "positive" constraint on G. In the following proof, G is an exponential function and satisfies this constriant. We need this constriant to show $G(|u|) \leq G(u) + G(-u)$.

Proof. (sketch) First, define a function $G(u) = e^{\lambda u}$ for some $\lambda > 0$ which will be finally removed by maximizing over λ. Use the log-exp trick and Jensen's inequality to insert G into the Rademacher average. Then apply Lemma 1 recursively. Finally using Lemma 2 and then optimizing the result over λ gives the result. □

Compared with the previous bound we have derived, although both are linear with regard to M, the latter has square root order instead of exponential order with regard to ℓ, which makes it much tighter.

References

1. Noah Golowich, Alexander Rakhlin, and Ohad Shamir. Size-independent sample complexity of neural networks. *arXiv preprint arXiv:1712.06541*, 2017.