Mathematics of Deep Learning
Lecture 9: Lazy Training and Generalization Bounds with Rademacher Complexity

Prof. Predrag R. Jelenković
Time: Tuesday 4:10-6:40pm

Dept. of Electrical Engineering
Columbia University , NY 10027, USA
Office: 812 Schapiro Research Bldg.
Phone: (212) 854-8174
Email: predrag@ee.columbia.edu
URL: http://www.ee.columbia.edu/∼predrag

Lazy Training

The following paper pinpoints the key ingredient responsible for wights remaining close to the initial values during training

- A Note on Lazy Training in Supervised Differentiable Programming, by L. Chizat and F. Bach, Dec 2018.

  See also an updated version:
  On Lazy Training Differentiable Programming, by L. Chizat and F. Bach, NIPS 2019.

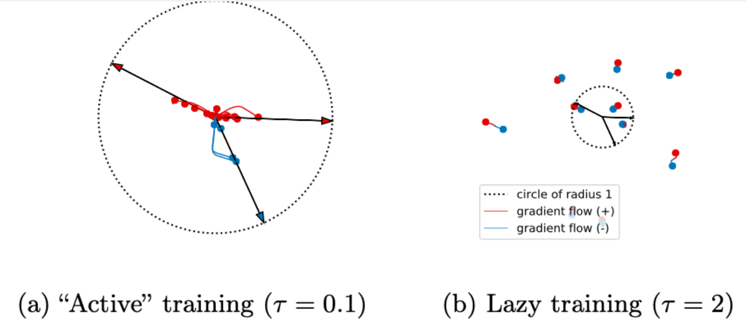
This paper puts in perspective many of the papers that we covered recently:

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In Advances in neural information processing systems, 2018.
Simon S. Du, Xiyu Zhai, Barnabs Pczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. In International Conference on Learning Representations, 2019.
Sanjeev Arora, Simon S. Du, Wei Hu, Zhiyuan Li, Ruslan Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. arXiv preprint arXiv:1904.11955, 2019.
Simon S. Du, Lee Jason D., Li Haochuan, Wang Liwei, and Zhai Xiyu. Gradient descent finds global minima of deep neural networks. In International Conference on Machine Learning (ICML), 2019.
Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. In Advances in Neural Information Processing Systems, pages 81678176, 2018.
Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In Proceedings of the 36th International Conference on Machine Learning, volume 97, pages 242252, 2019. Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Stochastic gradient descent optimizes over-parameterized deep ReLU networks. Machine Learning Journal, 2019.

The Answer Is In Scaling

They show that the key part of the model which ensures that the weights do not move far from the initial position $(w_{ij}(0))$ is the scaling $\alpha = 1/\sqrt{m}$, where m is the width of the network.

- Hence, the key to NTK approximation works because of scaling, rather than over-parametrization.
- They argue: any model can be scaled so that weights do not move much
  Example: τ - variance of $w_{ij}(0)$; "double trick" $f(x, w(0)) = 0$
  Ground truth: x uniform on a unit sphere, y output of NN with 3 neurons
  Model: $m = 20$ neurons - not a wide network

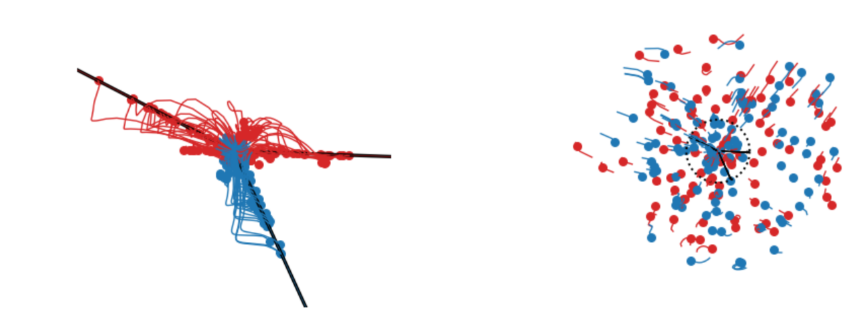


(a) "Active" training ($\tau = 0.1$)

(b) Lazy training ($\tau = 2$)

Many Neurons Example

- Ground truth: x uniform on a unit sphere, y output of NN with 3 neurons
- $n = 200$ data samples
- Model wide network: $m = 200$ neurons
- τ - variance of $w_{ij}(0)$



(a) "Active" training ($\tau = 0.1$)

(b) Lazy ($\tau = 2$)

Hence, the key to NTK regime is the scaling, rather than over-parametrization.

Recall Tangent Model

Consider any parametric model $f(w,x), w \in \mathbb{R}^p, x \in \mathbb{R}^d$. Then, assuming f is differentiable, we can linearize this model around the initial parameters w_0 as:

$$f_0(w,x) := f(w_0,x) + (w-w_0) \cdot \nabla_w f(w_0,x), \quad \text{(Tangent model)}. \quad (1)$$

Consider n data points $(x_i,y_i) \in \mathbb{R}^{d+1}, 1 \leq i \leq n$, and assume a quadratic loss, then training $f_0(w,x)$ is equivalent to solving a dual problem in the dot-product space according to tangent kernel (NTK):

$$K(x,y) = \nabla_w f(w_0,x) \cdot \nabla_w f(w_0,y).$$

Question: When is training $f_0(w,x)$ going to produce approximately the same results as $f(w,x)$?
Obviously, if $f(w,x)$ is linear, then $f_0(w,x) = f(w,x)$.

When Does Lazy Training Occur?

In other words, when can we expect that the trained $\boldsymbol{w}^*$ close to the initial $\boldsymbol{w}_0$, $\boldsymbol{w}^* \approx \boldsymbol{w}_0$.

- Let $F(\boldsymbol{w}) := L(f(\boldsymbol{w}))$, where L is the loss function. e.g., quadratic.
- Assume $F(\boldsymbol{w}_0) > 0$, $\nabla F(\boldsymbol{w}_0) \neq 0$, and consider a GD step

$$\boldsymbol{w}_1 = \boldsymbol{w}_0 - \eta \nabla F(\boldsymbol{w}_0).$$

- We can expect lazy training when the differential of f does not change much while the loss changes significantly, i.e.,

$$\begin{aligned}
\Delta(F) := \frac{|F(\boldsymbol{w}_1) - F(\boldsymbol{w}_0)|}{F(\boldsymbol{w}_0)} &\gg \Delta(Df) := \frac{\|Df(\boldsymbol{w}_1) - Df(\boldsymbol{w}_0)\|}{\|Df(\boldsymbol{w}_0)\|} \\
\Delta(F) \approx \eta \frac{\|\nabla F(\boldsymbol{w}_0)\|^2}{F(\boldsymbol{w}_0)} &\gg \Delta(Df) \approx \eta \frac{\|\nabla F(\boldsymbol{w}_0)\| \|D^2 f(\boldsymbol{w}_0)\|}{\|Df(\boldsymbol{w}_0)\|} \\
\frac{\|\nabla F(\boldsymbol{w}_0)\|}{F(\boldsymbol{w}_0)} &\gg \frac{\|D^2 f(\boldsymbol{w}_0)\|}{\|Df(\boldsymbol{w}_0)\|}
\end{aligned}$$

Df is a Jacobian, and $D^2 f$ is a Hessian matrix.

When Does Lazy Training Occur?

- For quadratic function $L(f) = \|f - y\|^2/2$, the previous condition simplifies to

$$\kappa_f(\boldsymbol{w}_0) := \|f(\boldsymbol{w}_0) - y\| \frac{\|D^2 f(\boldsymbol{w}_0)\|}{\|Df(\boldsymbol{w}_0)\|^2} \ll 1$$

  using the approximation

$$\|\nabla F(\boldsymbol{w}_0)\| = \|Df(\boldsymbol{w}_0)^\top (f(\boldsymbol{w}_0) - y)\| \approx \|Df(\boldsymbol{w}_0)\| \|f(\boldsymbol{w}_0) - y\|$$

- $\kappa_f(\boldsymbol{w}_0)$ could be called **relative scale** of model f at $\boldsymbol{w}_0$.

Lazy Training Examples

- **Rescaled model.** Consider a rescaled quadratic loss function

  $$\frac{\|\alpha f - y\|^2}{2\alpha^2},$$

  which yields

  $$\begin{aligned}\kappa_{\alpha f}(\boldsymbol{w}_0) &= \frac{1}{\alpha}\|\alpha f(\boldsymbol{w}_0) - y\| \frac{\|D^2 f(\boldsymbol{w}_0)\|}{\|Df(\boldsymbol{w}_0)\|^2} \\ &= O\left(\frac{1}{\alpha}\right), \quad (\text{if } \|\alpha f(\boldsymbol{w}_0) - y\| = O(1)) \\ &\ll 1, \quad (\text{as } \alpha \uparrow \infty)\end{aligned}$$

- $\|\alpha f(\boldsymbol{w}_0) - y\| = O(1))$ can be ensured by setting $f(\boldsymbol{w}_0) = 0$, which in NNs can be attained using a "doubling trick", i.e., repeating each neuron in the output layer with a negative linear weight.
- Hence, NN can be trained in the lazy regime even if networks are not wide. Large α is equivalent to large variance for $\boldsymbol{w}_0$.

Lazy Training Examples

- **One hidden layer NN.** Consider

$$f(x) = \alpha(m) \sum_{j=1}^{m} a_j \sigma(w_j \cdot x)$$
$$=: \alpha(m) \sum_{j=1}^{m} \phi(\theta_j, x),$$

  where $\theta_j = (a_j, w_j), \alpha(m) > 0, \mathbb{E}\phi(\theta_j, x) = 0$. Recall that the papers we considered used $\alpha(m) = 1/\sqrt{m}$, $\sigma(x) = x^+$ and $a_j \in \{0, 1\}$.

- Since $\mathbb{E}\phi(\theta_j, x) = 0$,

$$\mathbb{E}\|f(\boldsymbol{w}_0, x)\|^2 = m\alpha(m)^2 \mathbb{E}\|\phi(\theta)\|^2.$$

- For the differential, using the law of large numbers,

$$\frac{1}{m\alpha(m)^2} Df(\boldsymbol{w}_0) Df(\boldsymbol{w}_0)^\top = \frac{1}{m} \sum_{j=1}^{m} D\phi(\theta_j) D\phi(\theta_j)^\top \rightarrow \mathbb{E}[D\phi(\theta) D\phi(\theta)^\top]$$

Lazy Training Examples

- Hence

$$\mathbb{E}\|Df(\boldsymbol{w}_0)\|^2 = \mathbb{E}[Df(\boldsymbol{w}_0)Df(\boldsymbol{w}_0)^\top] \sim m\alpha(m)^2\mathbb{E}[D\phi(\theta)D\phi(\theta)^\top]$$

- Also, we can estimate the operator norm of Df as

$$\begin{aligned}\|D^2f(\boldsymbol{w}_0)\| &= \sup_{u\in\mathbb{R}^{dm},\|u\|\le 1} \alpha(m)\sum_{j=1}^{m} u_j^\top D^2\phi(\theta_j)u_j \\ &\le \alpha(m)\sup_{\theta_j}\|D^2\phi(\theta_j)\| \le \alpha(m)\mathsf{Lip}(D\phi),\end{aligned}$$

  where $\mathsf{Lip}(D\phi)$ is the Lipschitz constant of $D\phi$.

- Using the above and
  $\|f(\boldsymbol{w}_0) - y\| \le \|f(\boldsymbol{w}_0)\| + \|y\| = O(\alpha\sqrt{m}) + O(\sqrt{n})$

$$\mathbb{E}[\kappa_f(\boldsymbol{w}_0)] \approx O\left((\alpha\sqrt{m}) + \sqrt{n})\frac{\alpha}{\alpha^2 m}\right) = O(m^{-1/2} + (m\alpha(m))^{-1})$$

Lazy Training Examples

Hence, if $\alpha m \to \infty$, we have a Lazy Training regime.

In particular, in wide networks we can consider two interesting regimes

$$\alpha(m) = \frac{1}{\sqrt{m}} \Rightarrow \mathbb{E}[\kappa_f(\boldsymbol{w}_0)] = O(m^{-1/2}) \ll 1, \quad \text{(NTK/lazy regime)}$$
$$\alpha(m) = \frac{1}{m} \Rightarrow \mathbb{E}[\kappa_f(\boldsymbol{w}_0)] = O(1), \quad \text{(Mean field regime)}$$

In mean field regime

- The weights $\boldsymbol{w}$ move considerably during trining
- Much harder problem: requires extending gradient flow/descent to infinite dimensions
  - We might consider in the future some papers in this regime

Result On Lazy Training With General α-Scaling

Theorem 3.2 from Chizat and Bach (2019). This presentation follows Section 8 in Telgarsky's note. Notation

- Consider n data points $(x_i, y_i), 1 \leq i \leq n, x_i \in \mathbb{R}^d, y_i \in \mathbb{R}$, and let
$$f(w) = (f(x_1; w), \dots, f(x_n; w))^\top \in \mathbb{R}^n.$$
- Assume quadratic loss
$$L(\alpha f(w)) := \frac{1}{2}\|\alpha f(w) - y\|^2, \quad \alpha > 0.$$
- Then, the gradient flow evolves as
$$\dot{w}(t) = -\nabla_w L(\alpha f(w(t))) = -\alpha J_t^\top \nabla L(\alpha f(w(t))),$$
where $J_t = Df(w(t))$ is the Jacobian
$$J_t = J_{w(t)} = (\nabla f(x_1; w(t)), \dots, \nabla f(x_1; w(t)))^\top \in \mathbb{R}^{n \times p}$$
- Linear tangent model flow, with the same initialization $w(0)$
$$f_0(u) = f(w(0)) + J_0(u - w(0))$$
$$\dot{u}(t) = \nabla_w L(\alpha f_0(u(t))) = -\alpha J_0^\top \nabla L(\alpha f_0(u(t)))$$

Overparametrized NTK/Lazy Regime

- How close is the nonlinear gradient flow to the linear one?
- Assumptions

$$
\begin{aligned}
&\mathrm{rank}(J_0) = n \\
&\sigma_{\min} = \sigma_{\min}(J_0) = \sqrt{\lambda_{\min}(J_0 J_0^\top)} > 0 \\
&\|J_w - J_v\| \le \beta \|w - v\| \\
&\alpha \ge \frac{\beta \sigma_{\max} \sqrt{1152 L_0}}{\sigma_{\min}^3}, \quad L_0 = \frac{1}{2}\|\alpha f(w(0)) - y\|^2
\end{aligned}
$$

Theorem 8.1 (Telgarsky, also Theorem 3.2 in Chizat&Bach, 2019)
Under the preceding assumptions,

$$
\max(L(\alpha f(w(t))), L(\alpha f_0(u(t)))) \le L_0 \exp(-t\alpha^2 \sigma_{\min}^2/2)
$$

$$
\max(\|w(t) - w(0)\|, \|u(t) - w(0)\|) \le \frac{3\sigma_{\max}\sqrt{8L_0}}{\alpha \sigma_{\min}^2}
$$

- Simplifying the statement: computing the ballpark for $\beta, \sigma_{\min}, \sigma_{\max}, L_0, \alpha$ for shallow network.

Smoothness (Lipschitz) Constant β

- The order of Lipschitz constant β : $\|J_w - J_v\| \leq \beta \|w - v\|$
- Assume σ is β_0 smooth: $|\sigma'(w_j \cdot x) - \sigma'(v_i \cdot x)| \leq \beta_0 \|x\| \|w_j - v_j\|$
- Let $X \in \mathbb{R}^{n \times d}$ be a matrix with n training data rows. Then

$$
\begin{aligned}
\|J_w - J_v\|_2^2 &\leq \sum_{i,j} \|x_i\|^2 (\sigma'(w_j \cdot x_i) - \sigma'(v_i \cdot x_i))^2 \\
&\leq \sum_{i,j} \|x_i\|^4 \beta_0^2 \|w_j - v_j\|^2 \\
&\leq \beta_0^2 \|X\|_F^4 \|w - v\|^2
\end{aligned}
$$

- Hence, $\beta \leq \beta_0 \|X\|_F^2 = \Theta(n)$ (ballpark, assuming d is fixed)

Singular Values: $\sigma_{\min}, \sigma_{\max}$

- Scaling of singular values as a function of network width m
- By definition

$$(J_0 J_0^\top)_{i,j} = \nabla f(x_i; w(0))^\top \nabla f(x_j; w(0))$$

- If $w_j(0)$ are i.i.d. copies of v, then

$$\begin{aligned}\mathbb{E}(J_0 J_0^\top)_{i,j} &= \mathbb{E}[\nabla f(x_i; w(0))^\top \nabla f(x_j; w(0))] \\ &= \mathbb{E} \sum_k \sigma'(w_k(0) \cdot x_i) \sigma'(w_k(0) \cdot x_j) x_i \cdot x_j \\ &= m \mathbb{E} \sigma'(v \cdot x_i) \sigma'(v \cdot x_j) x_i \cdot x_j\end{aligned}$$

- Thus, one can expect that eigenvalues of $J_0 J_0^\top$ scale as m
- Implying that $\sigma_{\min}, \sigma_{\max}$ should scale as $\sqrt{m}$

Initial Risk: L_0

- Recall that (as in Du et. al)

$$f(x) = \sum_{j=1}^{m} a_j \sigma(w_j \cdot x)$$

  with $a_j \in \{-1, 1\}$. Hence, by the Central Limit Theorem, $f(x_i) = \Theta(\sqrt{m})$

- Implying

$$\sum_{i=1}^{n} (\alpha f(x_i))^2 = \Theta(\alpha^2 mn)$$

- Therefore

$$L_0 = \frac{1}{2} \sum_{i=1}^{n} (\alpha f(x_i) - y_i)^2 = \Theta(\alpha^2 mn)$$

  assuming $y_i = O(1)$ and $\alpha^2 m = \Omega(1)$

Combining All Parmeters

- Case: $L_0 = \Theta(\alpha^2 mn)$
- Using $\beta = \Theta(n), L_0 = \Theta(\alpha^2 mn)$ and the condition on α

  $$\alpha \geq \frac{\beta\sigma_{\max}\sqrt{1152L_0}}{\sigma_{\min}^3} \approx \frac{\beta\sigma_{\max}\alpha\sqrt{mn}}{\sigma_{\min}^3}$$

  implies

  $$\sigma_{\min}^3 \geq \beta\sigma_{\max}\sqrt{mn} \approx \sigma_{\max}\sqrt{mn^3}$$

  and, since $\sigma_{\min}, \sigma_{\max}$ are of the order $\sqrt{m}$,

  $$m^{3/2} \leq \sqrt{m^2n^3} \Rightarrow m \geq n^3$$

- Let's see how this simplifies the bounds in main theorem.

Combining All Parmeters

Recall $\sigma_{\max}$ and $\sigma_{\min}$ are both $\Theta(\sqrt{m})$, and assume $\alpha = \Theta(1/\sqrt{m})$, then

$$\begin{aligned}\max(L(\alpha f(w(t))), L(\alpha f_0(u(t)))) &\leq L_0 \exp(-t\alpha^2\sigma_{\min}^2/2) \\ &= O\left(\alpha^2 nm \exp(-t\alpha^2\sigma_{\min}^2/2)\right) \\ &= O\left(\alpha^2 nm \exp(-tc\alpha^2 m/2)\right) \\ &= O\left(n \exp(-\Omega(t))\right),\end{aligned}$$

as seen in Chizat&Bach.

$$\begin{aligned}\max(\|w(t) - w(0)\|, \|u(t) - w(0)\|) &\leq \frac{3\sigma_{\max}\sqrt{8L_0}}{\alpha\sigma_{\min}^2} \\ &= O\left(\frac{\sigma_{\max}\sqrt{\alpha^2 mn}}{\alpha\sigma_{\min}^2}\right) \\ &= O\left(\sqrt{n}\right)\end{aligned}$$

The proof of the theorem can be found in Section 8.1 in Telgarsky's monograph, 2021.

Bounding Generalization Error

The main objective of statistical learning is to predict well on future data.

- How do we measure the error?
  - Regression: Quadratic error/loss

    $$\ell(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$$

  - Classification: $\ell(\hat{y}, y) = 1_{\{\hat{y} \neq y\}}$
- $\{\mathcal{H}\}$: Hypothesis class of functions, e.g., all functions, $h(x, w)$, that can be generated by a NN of a certain architecture.
- Empirical Risk/Loss - total training error: For a data sample $S = \{(x_i, y_i)\}_{i=1}^n$ and $h \in \mathcal{H}$

  $$\hat{L}_n = \hat{L}_n(h) \equiv L_S(h) = \frac{1}{n}\sum_{i=1}^{n} \ell(h(x_i), y_i)$$

  Shai's book [ML2014] uses $L_S(h)$ notation; we'll use these interchangeably.

Empirical Risk Minimization and True Error

During training one typically minimizes the empirical risk (ERM), and obtain $\hat{h}_n \equiv h_S$

$$\hat{h}_n \in \arg\min_{h \in \mathcal{H}} \hat{L}(h)$$

True Risk=Population Risk

- Let $x \in \mathcal{X}, y \in \mathcal{Y}$, say $\mathcal{X} \subset \mathbb{R}^d, \mathcal{Y} \subset \mathbb{R}$, $z = (x, y) \in \mathcal{Z} = \mathcal{X} \times \mathcal{Y}$
- Define a probability measure on $\mathcal{Z}$, denoted by $\mathcal{D}$ in [ML2014] book
- Let $\{z_i = (x_i, y_i)\}_{i=1}^n$ be i.i.d. random variables on a product probability space $\mathcal{Z}^n$.
- Then, for $h \in \mathcal{H}$, we define a true risk/population risk as

$$L(h) \equiv L_{\mathcal{D}}(h) := \mathbb{E}_{x,y}\ell(h(x), y)$$

Probably Approximately Correct (PAC) Learning

The ultimate goal is to find h that minimizes the true risk, i.e.,

$$h \in \arg\min_{h \in \mathcal{H}} L(h)$$

But this is often impossible, leading to the more relaxed definition

> **Definition** (PAC Learnability) A hypothesis class $\mathcal{H}$ is (agnostic) PAC learnable with respect to a set $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ and a loss function $\ell : \mathcal{H} \times \mathcal{Z} \to \mathbb{R}_+$, if there exists a learning algorithm, which returns $\hat{h} \equiv \hat{h}(S)$ for a sample S of size $|S| = m$ with the following property: for every $0 < \epsilon, \delta < 1$ there exists $m(\epsilon, \delta)$, such that for all $m \geq m(\epsilon, \delta)$,
>
> $$\mathbb{P}\left[\left\{S \in \mathcal{Z}^m \;:\; L(\hat{h}(S)) \leq \min_{h \in \mathcal{H}} L(h) + \epsilon\right\}\right] \;\geq\; 1 - \delta$$

- PAC learning was introduced by Leslie Valiant (1984), who won for it the Turing Award in 2010.

Uniform Convergence is Sufficient for Learnability

Definition (uniform convergence)

$\mathcal{H}$ has the *uniform convergence property* if there exists $m(\epsilon, \delta)$, such that for every $m \geq m(\epsilon, \delta)$, every distribution $\mathcal{D}$, and any i.i.d. sample $S \equiv S_m$ with distribution $\mathcal{D}$, S is ϵ-representatve with probability $1 - \delta$, i.e., if $m \geq m(\epsilon, \delta)$

$$\mathbb{P}\left[\left\{S \in \mathcal{Z}^m : \sup_{h \in \mathcal{H}} |\hat{L}_S(h) - L(h)| \leq \epsilon\right\}\right] \geq 1 - \delta$$

- A direct consequence of the preceding definition and lemma:

Corollary

- *If $\mathcal{H}$ has the uniform convergence property with a function $m(\epsilon, \delta)$ then $\mathcal{H}$ is PAC learnable with the sample complexity $m(\epsilon, \delta) \leq m(\epsilon/2, \delta)$.*
- *Furthermore, in that case, the $\mathrm{ERM}_{\mathcal{H}}$ paradigm is a successful agnostic PAC learner for $\mathcal{H}$.*

Finite Classes are PAC Learnable

Uniform convergence property is really important. Recall that is how we proved the finite classes case.

Theorem

Assume $\mathcal{H}$ is finite and the range of the loss function is $[0, 1]$. Then, $\mathcal{H}$ is agnostically PAC learnable using the $\mathrm{ERM}_{\mathcal{H}}$ algorithm with sample complexity

$$m_{\mathcal{H}}(\epsilon, \delta) \leq \left\lceil \frac{2\log(2|\mathcal{H}|/\delta)}{\epsilon^2} \right\rceil .$$

In proving this theorem, we established a uniform convergence property

$$\mathbb{P}[\sup_{h\in\mathcal{H}} |\hat{L}_S(h) - L(h)| > \epsilon] \; \leq \; 2\,|\mathcal{H}|\,\exp\left(-2\,m\,\epsilon^2\right)$$

So, if $m \geq \frac{\log(2|\mathcal{H}|/\delta)}{2\epsilon^2}$ then the right-hand side is at most δ as required.

The Discretization Trick

- Suppose $\mathcal{H}$ is parameterized by d numbers
- Suppose we are happy with a representation of each number using b bits (say, $b = 32$)
- Then $|\mathcal{H}| \leq 2^{db}$, and so

$$m_{\mathcal{H}}(\epsilon, \delta) \leq \left\lceil \frac{2db + 2\log(2/\delta)}{\epsilon^2} \right\rceil .$$

- While not very elegant, it's a great tool for upper bounding sample complexity

Learning Infinite Hypothesis Classes

For most hypothesis classes $|\mathcal{H}| = \infty$: What can be done here?

Common measures of complexity of hypothesis classes

- Vapnik-Chervonenkis (VC) dimension
  Chapter 6 in Shai's [ML2014] book
- Rademacher Complexity
  All results are from Chapter 26 in [ML2014] book
  this was recently used in
  - A Priori Estimates For Two-layer Neural Networks, by Weinan et al., Jan 2019.
  - Fine-Grained Analysis of Optimization and Generalization for Overparameterized Two-Layer Neural Networks, by Arora et al., Jan 2019.
- PAC - Bayes: Chapter 31 in [ML20014]
  used recently in several NN papers, e.g.: see Neyshabur et al.
- Compression Bounds: Chapter 30 in [ML2014]

Rademacher Complexity: Motivation

- To simplify the notation, let $f(z) \equiv f(h,z) = \ell(h,z)$ and let $\mathcal{F}$ be the set of these functions
- Then, for $f \in \mathcal{F}$, the population/true risk end empirical risk are equal to

$$L(f) = \mathbb{E}_z[f(z)], \qquad \hat{L}_S(f) = \frac{1}{m}\sum_{i=1}^{m} f(z_i)$$

- Recall ϵ-representative sample: A training set S is called ϵ-representative if

$$\forall h \in \mathcal{H}, \qquad |\hat{L}_S(h) - L(h)| \leq \epsilon \ .$$

- Which motivates the following definition

Definition *Representativeness of S*: is the largest gap between the true error and empirical error

$$\text{Rep}(\mathcal{F}, S) = \sup_{f \in \mathcal{F}} |\hat{L}_S(f) - L(f)|$$

Rademacher Complexity: Motivation

- But we don't know the true error $L(f)$
- Replace $L(f)$ by another empirical error
- Partition the training data S into two disjoint sets: S_1, S_2 and estimate the representativeness of S by

$$\sup_{f \in \mathcal{F}} (\hat{L}_{S_1}(f) - \hat{L}_{S_2}(f))$$

- Let $\sigma_i = 1$ if $z_i \in S_1$ and $\sigma_i = -1$, otherwise. Then

$$\sup_{f \in \mathcal{F}} (\hat{L}_{S_1}(f) - \hat{L}_{S_2}(f)) = \frac{1}{m} \sup_{f \in \mathcal{F}} \sum_{i=1}^{m} \sigma_i f(z_i)$$

  which motivates the following definition

Definition Let σ_i be i.i.d. with $\mathbb{P}[\sigma_i = 1] = \mathbb{P}[\sigma_i = -1] = 1/2$. Then, the *Rademacher Complexity* of $\mathcal{F}$ with respect to sample S is defined as

$$R(\mathcal{F}, S) := \frac{1}{m} \mathbb{E}_\sigma \sup_{f \in \mathcal{F}} \sum_{i=1}^{m} \sigma_i f(z_i)$$

Rademacher Complexity

Lemma 26.2 The expected value of the representativeness of S is bounded by

$$\mathbb{E}_S[\mathsf{Rep}(\mathcal{F}, S)] \equiv \mathbb{E}_S \sup_{f \in \mathcal{F}} |\hat{L}_S(f) - L(f)| \leq 2\mathbb{E}_S[R(\mathcal{F}, S)]$$

Proof: Let $S' = \{z'_i\}_{i=1}^m$ be another i.i.d. sample independent of S. Then

$$L(f) - \hat{L}_S(f) = \mathbb{E}_{S'}[\hat{L}_{S'}(f) - \hat{L}_S(f)]$$

which implies (note $\hat{L}_{S'}(f) - \hat{L}_S(f) \leq \sup_{f \in \mathcal{F}}(\hat{L}_{S'}(f) - \hat{L}_S(f))$)

$$\sup_{f \in \mathcal{F}}(L(f) - \hat{L}_S(f)) = \sup_{f \in \mathcal{F}} \mathbb{E}_{S'}[\hat{L}_{S'}(f) - \hat{L}_S(f)] \leq \mathbb{E}_{S'}\left[\sup_{f \in \mathcal{F}}(\hat{L}_{S'}(f) - \hat{L}_S(f))\right]$$

Rademacher Complexity

Proof: (continued) Taking the expectation over S

$$\begin{aligned}
\mathbb{E}_S\left[\sup_{f\in\mathcal{F}}(L(f)-\hat{L}_S(f))\right] &\leq \mathbb{E}_{S,S'}\left[\sup_{f\in\mathcal{F}}(\hat{L}_{S'}(f)-\hat{L}_S(f))\right] \\
&= \frac{1}{m}\mathbb{E}_{S,S'}\left[\sup_{f\in\mathcal{F}}\sum_{i=1}^{m}(f(z_i')-f(z_i))\right] \\
&= \frac{1}{m}\mathbb{E}_{S,S'}\left[\sup_{f\in\mathcal{F}}\left(f(z_j')-f(z_j)+\sum_{i\neq j}(f(z_i')-f(z_i))\right)\right] \\
&= \frac{1}{m}\mathbb{E}_{S,S'}\left[\sup_{f\in\mathcal{F}}\left(f(z_j)-f(z_j')+\sum_{i\neq j}(f(z_i')-f(z_i))\right)\right] \\
&= \frac{1}{m}\mathbb{E}_{S,S',\sigma_j}\left[\sup_{f\in\mathcal{F}}\left(\sigma_j(f(z_j')-f(z_j))+\sum_{i\neq j}(f(z_i')-f(z_i))\right)\right] \\
&= \frac{1}{m}\mathbb{E}_{S,S',\sigma}\left[\sup_{f\in\mathcal{F}}\left(\sum_{i}\sigma_i(f(z_i')-f(z_i))\right)\right]
\end{aligned}$$

Rademacher Complexity

Proof: (continued) Taking the expectation over S

$$\begin{aligned}
\mathbb{E}_S \left[\sup_{f \in \mathcal{F}} (L(f) - \hat{L}_S(f)) \right] &\leq \frac{1}{m} \mathbb{E}_{S,S',\sigma} \left[\sup_{f \in \mathcal{F}} \left(\sum_i \sigma_i (f(z_i') - f(z_i)) \right) \right] \\
&\leq \frac{1}{m} \mathbb{E}_{S,S',\sigma} \left[\sup_{f \in \mathcal{F}} \left(\sum_i \sigma_i f(z_i') \right) + \sup_{f \in \mathcal{F}} \left(\sum_i -\sigma_i f(z_i)) \right) \right] \\
&= 2\mathbb{E}_S[R(\mathcal{F}, S)]
\end{aligned}$$

since σ_i and $-\sigma_i$ have the same distribution.

Need a More General Inequality

Generalizations of Hoeffding's Inequality:

- Azuma'a Inequality
  - Generalization to martingales with bounded differences
- McDiarmid's Inequality: the one we need
  - Basically a corollary of Azuma's
  - Can be proved directly using the same proof as for Azuma's inequality
  - The proof is a combination of martingale arguments and the ones we used for Hoeffding's inequality

McDiarmid's Inequality

Lemma

McDiarmid's Inequality Consider independent random random variables $X_1, X_2, \dots, X_n \in \mathcal{X}$ *and a function* $f : \mathcal{X}^n \to \mathbb{R}$*. If for all* $i \in \{1, \dots, n\}$ *and all* $x_1, \dots, x_n, x_i' \in \mathcal{X}$*, the function* f *satisfies*

$$|f(x_1, \dots, x_{i-1}, x_i, x_{i+1} \dots, x_n) - f(x_1, \dots, x_{i-1}, x_i', x_{i+1} \dots, x_n)| \leq c,$$

then

$$\mathbb{P}[|f(X_1, \dots, X_n) - \mathbb{E}[f(X_1, \dots, X_n)]| > t] \leq 2 \exp\left(-\frac{2t^2}{nc^2}\right)$$

or, equivalently, with probability at least $1 - \delta$*,*

$$|f(X_1, \dots, X_n) - \mathbb{E}[f(X_1, \dots, X_n)]| \leq c\sqrt{\frac{2}{n} \log\left(\frac{2}{\delta}\right)}.$$

McDiarmid's Inequality

Proof: To simplify the notation, let's denote $f \equiv f(X_1, \ldots, X_n)$. Note that $M_i = \mathbb{E}[f|X_1, \ldots, X_i], M_0 = \mathbb{E}[f]$, is a martingale and consider martingale differences

$$V_i = \mathbb{E}[f|X_1, \ldots, X_i] - \mathbb{E}[f|X_1, \ldots, X_{i-1}].$$

Next, observe that $\mathbb{E}[V_i] = 0$ and

$$\sum_{i=1}^{n} V_i = \mathbb{E}[f|X_1, \ldots, X_n] - \mathbb{E}[f] = f(X_1, \ldots, X_n) - \mathbb{E}[f].$$

Now, we show that the martingale differences, V_i, are uniformly bounded. (Recall, having bounded random variables was the key component of Hoeffding's proof.) To this end, define, for x in ith position

$$L_i = \inf_x \mathbb{E}[f(X_1, \ldots, x, \ldots, X_n)|X_1, \ldots, X_i] - \mathbb{E}[f|X_1, \ldots, X_{i-1}]$$
$$U_i = \sup_x \mathbb{E}[f(X_1, \ldots, x, \ldots, X_n)|X_1, \ldots, X_i] - \mathbb{E}[f|X_1, \ldots, X_{i-1}]$$

and note that

$$L_i \leq V_i \leq U_i$$

McDiarmid's Inequality

Proof: (continued) Next, we show that $U_i - L_i$ is uniformly bounded

$$\begin{aligned} U_i - L_i &= \sup_{x,x'} \left(\mathbb{E}[f(X_1, \ldots, x, \ldots, X_n) - f(X_1, \ldots, x', \ldots, X_n) | X_1, \ldots, X_{i-1}] \right) \\ &\leq c \end{aligned}$$

where the last equality is by the assumption on f and independence.

Hence, $V_i \in [L_i, L_i + c]$, where L_i is a function of $X_1, \ldots, X_{i-1}$.

Therefore, L_i can be treated as a constant with respect to the conditional expectation $\mathbb{E}[\cdot | X_1, \ldots, X_{i-1}]$

Meaning, we can use exactly the same arguments, convexity of $e^{\lambda x}$, to prove

$$\mathbb{E}\left[e^{\lambda V_i} | X_1, \ldots, X_{i-1} \right] \leq e^{\frac{\lambda^2 c^2}{8}}$$

McDiarmid's Inequality

Proof: (continued) Then, using the preceding inequality, we obtain

$$
\begin{aligned}
\mathbb{P}[f - \mathbb{E}[f] > t] &= \mathbb{P}[\sum_{i=1}^{n} V_i > t] \\
&\le e^{-\lambda t}\mathbb{E}\left[\prod_{i=1}^{n} e^{\lambda V_i}\right] \\
&\le e^{-\lambda t}\mathbb{E}\left[\prod_{i=1}^{n-1} e^{\lambda V_i}\mathbb{E}[e^{\lambda V_n}|X_1,\ldots,X_{n-1}]\right] \\
&\le e^{-\lambda t}e^{\frac{\lambda^2 c^2}{8}}\mathbb{E}\left[\prod_{i=1}^{n-1} e^{sV_i}\right] \\
&\le \cdots \le e^{-\lambda t}e^{\frac{n\lambda^2 c^2}{8}}
\end{aligned}
$$

Finally, the proof is completed by optimizing over λ, i.e., setting

$$\lambda = \frac{4t}{nc^2}.$$

Back to Generalization Bounds

Theorem 26.6 (Shais' book) Assume that for all z and $h \in \mathcal{H}$, we have $|\ell(h,z)| \leq c$, and recall $S = \{z_1, \ldots, z_m\}$ is the sample. Then, with probability at least $1-\delta$, for any $h \in \mathcal{H}$ (and in particular for $h = ERM_{\mathcal{H}}(S)$),

$$L(h) - \hat{L}_S(h) \leq 2\mathbb{E}_S[R(\mathcal{F}, S)] + c\sqrt{\frac{2}{m}\log\left(\frac{2}{\delta}\right)}$$

Proof: First, note that random variable

$$\text{Rep}(\mathcal{F}, S) = \sup_{h \in \mathcal{H}} (L(h) - \hat{L}_S(h))$$

satisfies the bounded difference condition of McDiarmid's lemma with a constant $2c/m$.

Combining the bound form Lemma 26.2 and McDiarmid's lemma, we obtain that with probability at least $1-\delta$,

$$\text{Rep}(\mathcal{F}, S) \leq \mathbb{E}[\text{Rep}(\mathcal{F}, S)] + c\sqrt{\frac{2}{m}\log\left(\frac{2}{\delta}\right)} \leq 2\mathbb{E}_S[R(\mathcal{F}, S)] + c\sqrt{\frac{2}{m}\log\left(\frac{2}{\delta}\right)}$$

Rademacher Calculus

Hence, to make the preceding theorem useful, we must find the ways to estimate

$$\mathbb{E}_S[R(\mathcal{F}, S)] = ?$$

With a small abuse of notation, from any set $A \in R^m$ let us define

$$R(A) := \frac{1}{m}\mathbb{E}_\sigma\left[\sup_{\boldsymbol{a}\in A}\sum_{i=1}^{m} a_i\sigma_i\right]$$

This lemma is immediate for the definition. (Why?)

Lemma 26.6 For any $A \subset \mathbb{R}^m$, $c \in \mathbb{R}$, and $\boldsymbol{a}_0 \in \mathbb{R}^m$, we have

$$R(\{c\boldsymbol{a} + \boldsymbol{a}_0 : \boldsymbol{a} \in A\}) \leq |c|R(A).$$

Contraction Lemma

Lemma 26.9 (Contraction Lemma) For each $i \in \{1, \dots, m\}$, let $\phi_i : \mathbb{R} \to \mathbb{R}$ be ρ-Lipschitz, i.e., for any $\alpha, \beta \in \mathbb{R}$, we have $\phi_i(\alpha) - \phi_i(\beta) \leq \rho|\alpha - \beta|$. Furthermore, for any $\in \mathbb{R}$, let $\boldsymbol{\phi}(\boldsymbol{a}) = (\phi_1(a_1), \dots, \phi_m(a_m))$ and $\boldsymbol{\phi} \circ A = \{\boldsymbol{\phi}(\boldsymbol{a}) : \boldsymbol{a} \in A\}$. Then,

$$R(\boldsymbol{\phi} \circ A) \leq \rho R(A).$$

Proof: Without loss of generality we can assume $\rho = 1$; if $\rho \neq 1$, define $\phi' = \phi/\rho$ and use the preceding Lemma 26.6.

Next, let $A_i = \{(a_1, \dots, a_{i-1}, \phi_i(a_i), a_{i+1}, \dots, a_m) : \boldsymbol{a} \in A\}$.

Clearly, it is enough to prove that

$$R(A_i) \leq R(A)$$

For simplicity, assume $i = 1$ and omit the subscript $\phi \equiv \phi_1$

Contraction Lemma

Proof: (continued)

$$
\begin{aligned}
mR(A_1) &= \mathbb{E}_\sigma \left[\sup_{\boldsymbol{a} \in A_1} \sum_{i=1}^{m} \sigma_i a_i \right] \\
&= \mathbb{E}_\sigma \left[\sup_{\boldsymbol{a} \in A} \left(\sigma_1 \phi(a_1) + \sum_{i=2}^{m} \sigma_i a_i \right) \right] \\
&= \frac{1}{2} \mathbb{E}_{\sigma_2, \ldots, \sigma_m} \left[\sup_{\boldsymbol{a} \in A} \left(\phi(a_1) + \sum_{i=2}^{m} \sigma_i a_i \right) + \sup_{\boldsymbol{a} \in A} \left(-\phi(a_1) + \sum_{i=2}^{m} \sigma_i a_i \right) \right] \\
&= \frac{1}{2} \mathbb{E}_{\sigma_2, \ldots, \sigma_m} \left[\sup_{\boldsymbol{a}, \boldsymbol{a}' \in A} \left(\phi(a_1) - \phi(a_1') + \sum_{i=2}^{m} \sigma_i a_i + \sum_{i=2}^{m} \sigma_i a_i' \right) \right] \\
&\leq \frac{1}{2} \mathbb{E}_{\sigma_2, \ldots, \sigma_m} \left[\sup_{\boldsymbol{a}, \boldsymbol{a}' \in A} \left(|a_1 - a_1'| + \sum_{i=2}^{m} \sigma_i a_i + \sum_{i=2}^{m} \sigma_i a_i' \right) \right]
\end{aligned}
$$

where in the last inequality we used LC of ϕ with $\rho = 1$.

Contraction Lemma

Proof: (continued) Since both $\boldsymbol{a}, \boldsymbol{a}' \in A$, we can omit the absolute value in $|a_1 - a_1'|$ (?), and thus

$$mR(A_1) \leq \frac{1}{2}\mathbb{E}_{\sigma_2,\ldots,\sigma_m}\left[\sup_{\boldsymbol{a},\boldsymbol{a}'\in A}\left(a_1 - a_1' + \sum_{i=2}^{m}\sigma_i a_i + \sum_{i=2}^{m}\sigma_i a_i'\right)\right]$$

$$= \mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\boldsymbol{a}\in A}\sum_{i=1}^{m}\sigma_i a_i\right] = mR(A),$$

which concludes the proof since we can iterate this over all indices $i \in \{1,\ldots,m\}$.

Massarat Lemma

Rademacher complexity of a finite set $A = \{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n\}, \boldsymbol{a}_i \in \mathbb{R}^m$, grows logarithmically with the size of the set $|A| = n$.

Lemma 26.8 (Massarat Lemma) Let $\bar{\boldsymbol{a}} = (1/n) \sum_{i=1}^n \boldsymbol{a}_i$. Then,

$$R(A) \leq \max_{\boldsymbol{a} \in A} \|\boldsymbol{a} - \bar{\boldsymbol{a}}\| \frac{\sqrt{2 \log(|A|)}}{m}.$$

Proof: By Lemma 26.6, we can assume $\bar{\boldsymbol{a}} = 0$.
Next, for $\lambda > 0$, let $A' = \lambda A$. We upper bound

$$\begin{aligned} mR(A') &= \mathbb{E}_{\boldsymbol{\sigma}} \left[\max_{\boldsymbol{a} \in A'} \langle \boldsymbol{\sigma}, \boldsymbol{a} \rangle \right] = \mathbb{E}_{\boldsymbol{\sigma}} \left[\log \left(\max_{\boldsymbol{a} \in A'} e^{\langle \boldsymbol{\sigma}, \boldsymbol{a} \rangle} \right) \right] \\ &\leq \mathbb{E}_{\boldsymbol{\sigma}} \left[\log \left(\sum_{\boldsymbol{a} \in A'} e^{\langle \boldsymbol{\sigma}, \boldsymbol{a} \rangle} \right) \right] \\ &\leq \log \left(\mathbb{E}_{\boldsymbol{\sigma}} \left[\sum_{\boldsymbol{a} \in A'} e^{\langle \boldsymbol{\sigma}, \boldsymbol{a} \rangle} \right] \right) \qquad \text{Jensen's inequality} \end{aligned}$$

Massarat Lemma

Proof: (continued)

$$mR(A') \leq \log\left(\sum_{\boldsymbol{a}\in A'}\prod_{i=1}^{m}\mathbb{E}_{\sigma_i}\left[e^{\sigma_i a_i}\right]\right)$$

Next,

$$\mathbb{E}_{\sigma_i}\left[e^{\sigma_i a_i}\right] = \frac{e^{a_i}+e^{-a_i}}{2} \leq e^{a_i^2/2}$$

and therefore

$$\begin{aligned} mR(A') &\leq \log\left(\sum_{\boldsymbol{a}\in A'}\prod_{i=1}^{m}e^{a_i^2/2}\right) = \log\left(\sum_{\boldsymbol{a}\in A'}\prod_{i=1}^{m}e^{\|\boldsymbol{a}\|^2/2}\right) \\ &\leq \log\left(|A'|\max_{\boldsymbol{a}\in A'}e^{\|\boldsymbol{a}\|^2/2}\right) = \log(|A'|) + \max_{\boldsymbol{a}\in A'}\|\boldsymbol{a}\|^2/2 \end{aligned}$$

Massarat Lemma

Proof: (continued) Since, by Lemma 26.6, $R(A) \le R(A')/\lambda$, we obtain

$$R(A) \le \frac{\log(|A|) + \lambda^2 \max_{\boldsymbol{a} \in A} \|\boldsymbol{a}\|^2/2}{\lambda m}$$

By choosing λ that achieves minimum in the expression above, i.e., $\lambda = \sqrt{2\log(|A|)/\max_{\boldsymbol{a} \in A} \|\boldsymbol{a}\|^2}$ and rearranging the terms, we conclude the proof.

Rademacher Complexity of Linear Classes

Linear class bounded by L_2 norm. Let

$$\mathcal{H}_2 = \{\boldsymbol{x} \to \langle \boldsymbol{w}, \boldsymbol{x} \rangle : \|\boldsymbol{w}\|_2 \leq 1\}$$

Lemma 26.10 Let $S = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ be vectors in $\boldsymbol{x}_i \in \mathbb{R}^n$. Define $\mathcal{H}_2 \circ S = \{\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle : \|\boldsymbol{w}\|_2 \leq 1, 1 \leq i \leq m\}$. Then,

$$R(\mathcal{H}_2 \circ S) \leq \frac{\max_i \|\boldsymbol{x}_i\|_2}{\sqrt{m}}.$$

Remark: Note that the bound does not depend on the dimension of $\boldsymbol{x}$.
Proof: Using Cauchy-Schwartz inequality $|\boldsymbol{x} \cdot \boldsymbol{y}| \leq \|\boldsymbol{x}\|_2 \|\boldsymbol{y}\|_2$,

$$\begin{aligned} mR(\mathcal{H}_2 \circ S) &= \mathbb{E}_\sigma \left[\sup_{\boldsymbol{a} \in \mathcal{H}_2 \circ S} \sum_{i=1}^m \sigma_i a_i \right] = \mathbb{E}_\sigma \left[\sup_{\boldsymbol{w} : \|\boldsymbol{w}\|_2 \leq 1} \sum_{i=1}^m \sigma_i \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle \right] \\ &= \mathbb{E}_\sigma \left[\sup_{\boldsymbol{w} : \|\boldsymbol{w}\|_2 \leq 1} \sum_{i=1}^m \langle \boldsymbol{w}, \sigma_i \boldsymbol{x}_i \rangle \right] \leq \mathbb{E}_\sigma \left[\left\| \sum_{i=1}^m \sigma_i \boldsymbol{x}_i \right\|_2 \right] \end{aligned}$$

Rademacher Complexity of Linear Classes

Proof: Next, using Jensen's inequality and concavity of $\sqrt{x}$

$$\mathbb{E}_\sigma \left[\left\| \sum_{i=1}^m \sigma_i \boldsymbol{x}_i \right\|_2 \right] = \mathbb{E}_\sigma \left[\left(\left\| \sum_{i=1}^m \sigma_i \boldsymbol{x}_i \right\|_2^2 \right)^{1/2} \right] \leq \left(\mathbb{E}_\sigma \left[\left\| \sum_{i=1}^m \sigma_i \boldsymbol{x}_i \right\|_2^2 \right] \right)^{1/2}$$

Next

$$\begin{aligned} \mathbb{E}_\sigma \left[\left\| \sum_{i=1}^m \sigma_i \boldsymbol{x}_i \right\|_2^2 \right] &= \mathbb{E}_\sigma \left[\sum_{i,j} \sigma_i \sigma_j \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle \right] \\ &= \sum_{i \neq j} \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle \mathbb{E}[\sigma_i \sigma_j] + \sum_{i=1}^m \langle \boldsymbol{x}_i, \boldsymbol{x}_i \rangle \mathbb{E}[\sigma_i^2] \\ &= \sum_{i=1}^m \|\boldsymbol{x}_i\|_2^2 \leq m \max_i \|\boldsymbol{x}_i\|_2^2. \end{aligned}$$

Rademacher Complexity of Linear Classes

Next, using Massart's lemma, we can derive a similar result for L_1 norm. Let

$$\mathcal{H}_1 = \{\boldsymbol{x} \to \langle \boldsymbol{w}, \boldsymbol{x} \rangle : \|\boldsymbol{w}\|_1 \leq 1\}$$

Lemma 26.11 Let $S = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ be vectors in $\boldsymbol{x}_i \in \mathbb{R}^n$. Then,

$$R(\mathcal{H}_1 \circ S) \leq \max_i \|\boldsymbol{x}_i\|_\infty \sqrt{\frac{2 \log(2n)}{m}}$$

Proof: Using $\langle \boldsymbol{w}, \boldsymbol{v} \rangle \leq \|\boldsymbol{w}\|_1 \|\boldsymbol{v}\|_\infty$,

$$\begin{aligned} mR(\mathcal{H}_1 \circ S) &= \mathbb{E}_\sigma \left[\sup_{\boldsymbol{a} \in \mathcal{H}_1 \circ S} \sum_{i=1}^m \sigma_i a_i \right] = \mathbb{E}_\sigma \left[\sup_{\boldsymbol{w} : \|\boldsymbol{w}\|_1 \leq 1} \sum_{i=1}^m \sigma_i \langle \boldsymbol{w}, \boldsymbol{x}_i \rangle \right] \\ &= \mathbb{E}_\sigma \left[\sup_{\boldsymbol{w} : \|\boldsymbol{w}\|_1 \leq 1} \sum_{i=1}^m \langle \boldsymbol{w}, \sigma_i \boldsymbol{x}_i \rangle \right] \leq \mathbb{E}_\sigma \left[\left\| \sum_{i=1}^m \sigma_i \boldsymbol{x}_i \right\|_\infty \right] \end{aligned}$$

Rademacher Complexity of Linear Classes

Proof: Next, for $1 \leq j \leq n$, let $\boldsymbol{v}_j = (x_{1,j}, \ldots, x_{m,j})$ be the jth coordinate of all $\boldsymbol{x}$ vectors. Note that

$$\|\boldsymbol{v}_j\|_2 \leq \sqrt{m} \max_i \|\boldsymbol{x}\|_\infty \quad \text{(why?)}$$

Now, let $V = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n, -\boldsymbol{v}_1, \ldots, -\boldsymbol{v}_n\}$. Then, observe

$$\mathbb{E}_\sigma \left[\left\| \sum_{i=1}^m \sigma_i \boldsymbol{x}_i \right\|_\infty \right] = mR(V) \leq m \max_i \|\boldsymbol{x}_i\|_\infty \sqrt{\frac{2\log(2n)}{m}},$$

where in the last inequality we used Massart's lemma.

Reading

NTK/Lazy training regime:

- A Note on Lazy Training in Supervised Differentiable Programming, by L. Chizat and F. Bach, Dec 2018.

  An updated version:
  On Lazy Training Differentiable Programming, by L. Chizat and F. Bach, NIPS 2019.

- Chapter 8 in the recent monograph Deep Learning Theory Lecture Notes, by Telgarsky, Feb 2021.

This paper puts in perspective many of the papers that we covered recently:

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In Advances in neural information processing systems, 2018.
Simon S. Du, Xiyu Zhai, Barnabs Pczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. In International Conference on Learning Representations, 2019.
Sanjeev Arora, Simon S. Du, Wei Hu, Zhiyuan Li, Ruslan Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. arXiv preprint arXiv:1904.11955, 2019.
Simon S. Du, Lee Jason D., Li Haochuan, Wang Liwei, and Zhai Xiyu. Gradient descent finds global minima of deep neural networks. In International Conference on Machine Learning (ICML), 2019.
Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. In Advances in Neural Information Processing Systems, pages 81678176, 2018.
Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In Proceedings of the 36th International Conference on Machine Learning, volume 97, pages 242252, 2019. Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Stochastic gradient descent optimizes over-parameterized deep ReLU networks. Machine Learning Journal, 2019.

Reading

Generalization bounds:

- PAC Learning and Generalization Theory - [ML2014] book:
  PAC learning: Chapters 2-4; Rademacher Complexity: Chapter 26
  (In particular, Theorem 26.5 and Lemmas 26.9 & 26.11)
  In general, for PAC learning theory see: Chapters: 2-6, 26-31
- Generalization bounds for NNs
  - A Priori Estimates For Two-layer Neural Networks, by Weinan et al., Jan 2019.
  - Fine-Grained Analysis of Optimization and Generalization for Overparameterized Two-Layer Neural Networks, by Arora et al., Jan 2019.
  - See the references in the preceding papers, and the citations on Google Scholar

Have Fun!