10707
Deep Learning: Spring 2021

Andrej Risteski

Machine Learning Department

Recitation 3:
Brief overview of VC and Rademacher bounds in deep learning

Classical view of generalization

Meta-"theorem" of generalization: with probability $1-\delta$ over the choice of a training set of size m, we have

$$\sup_{f \in \mathcal{F}} \left| \ \widehat{\mathbb{E}}\, l(f(x), y) - \mathbb{E}\, l(f(x), y) \right| \le O\left(\sqrt{\frac{\text{complexity}(\mathcal{F}) + \ln 1/\delta}{m}}\right)$$

Some measures of "complexity":

(Effective) number of elements in $\mathcal{F}$

VC (Vapnik-Chervonenkis)

Rademacher complexity

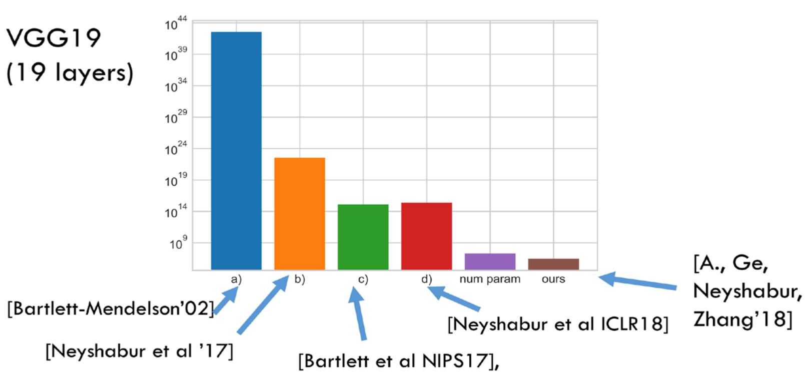
PAC-Bayes

What do these bounds look like?

VC-dimension of **fully connected net** with **N** edges in total is O(**N log N**).
Note: doesn't depends on magnitude of weights at all.

Rademacher bounds: a lot of recent activity [*Neyshabur et al '15, 17; Bartlett-Foster-Telgarsky '18, Golowich-Rakhlin-Shamir '19*]. Roughly, best bounds look like:

$$R_m = O\left(\frac{\sqrt{\Pi_{i=1}^{d} ||W_i||_2 \text{poly}(d)}}{\sqrt{m}}\right)$$, where d is the depth



When plugged in for standard values used in practice, the values these quantities give are **rarely "non-trivial"** (i.e. give a quantity < 1)

(Jiang et al'19): a large-scale investigation of the correlation of extant generalization measures with true generalization.

They show some newer Rademacher bounds have *worse* correlation.

Recap: VC dimension

Max # of possible label sequences

Let $\mathcal{F} = \{f: \mathcal{X} \to \{\pm 1\}\}$ be a class of predictors.

The **growth function** $\Pi_{\mathcal{F}}: \mathbb{N} \to \mathbb{N}$ of $\mathcal{F}$ is defined as

$$\Pi_{\mathcal{F}}(m) = \max_{(x_1, x_2, \dots x_m)} \left|\{\left(f(x_1), f(x_2), \dots, f(x_m)\right) \middle| f \in \mathcal{F}\}\right|$$

The **VC (Vapnis-Chervonenkis) dimension** of $\mathcal{F}$ is defined as

$$\mathrm{VCdim}(\mathcal{F}) = \max\{m\colon\ \Pi_{\mathcal{F}}(m) = 2^m\}$$

Equivalently: the largest m, s.t. $\mathcal{F}$ can **shatter** some set of size m:

that is, for **some** $(\mathrm{x}_1, \mathrm{x}_2, \dots x_m)$ and **any** labeling $(\mathrm{b}_1, \mathrm{b}_2, \dots b_m)$, $b_i \in \{\pm 1\}$,

some $\mathrm{f} \in \mathcal{F}$ can produce that labeling, that is

$$(f(\mathrm{x}_1), f(\mathrm{x}_2), \dots f(x_m)) = (\mathrm{b}_1, \mathrm{b}_2, \dots b_m)$$

Recap: VC dimension

Max # of possible label sequences

Let $\mathcal{F} = \{f : \mathcal{X} \to \{\pm 1\}\}$ be a class of predictors.

The **growth function** $\Pi_{\mathcal{F}} : \mathbb{N} \to \mathbb{N}$ of $\mathcal{F}$ is defined as

$$\Pi_{\mathcal{F}}(m) = \max_{(x_1, x_2, \ldots x_m)} \left|\left\{\left(f(x_1), f(x_2), \ldots, f(x_m)\right) \middle| f \in \mathcal{F}\right\}\right|$$

The **VC (Vapnis-Chervonenkis) dimension** of $\mathcal{F}$ is defined as

$$\mathrm{VCdim}(\mathcal{F}) = \max\{m :\ \Pi_{\mathcal{F}}(m) = 2^m\}$$

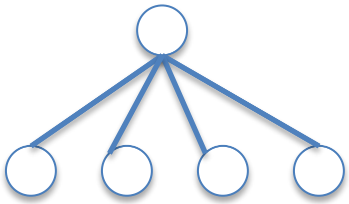
The two are closely related (**Sauer's lemma**):

$$\Pi_{\mathcal{F}}(m) = O\left(m^{\mathrm{VCdim}(\mathcal{F})}\right)$$

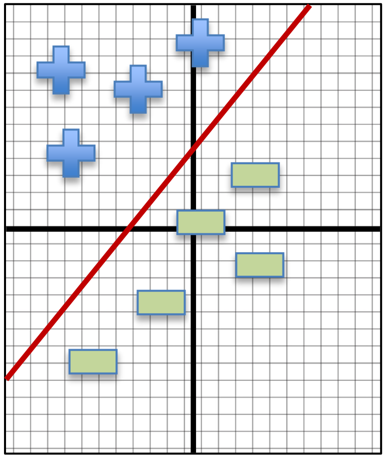
Bounding the VC dimension of a neural network

We'll prove a very simple bound on the VC dimension of a neural network with ***binary*** activation function. (Makes the proof the simplest).

$\operatorname{sgn}(\langle w, x\rangle + b)$



(Each unit calculates a "hyperplane" $\operatorname{sgn}(\langle w, x\rangle + b)$)



For starters, let's bound the VC dimension of a single unit, namely consider

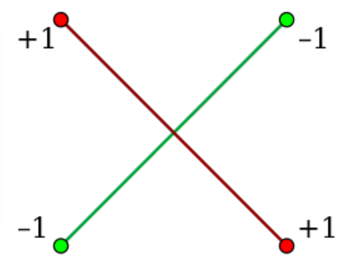
$\mathcal{F} = \{\operatorname{sgn}(\langle w, x\rangle + b) \mid w \in \mathbb{R}^d, b \in \mathbb{R}\}$.

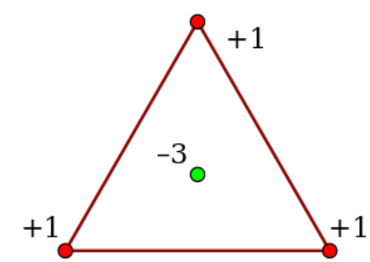
We'll show that $\operatorname{VCdim}(\mathcal{F}) \le d + 1$.

Bounding the VC dimension of a neural network

VCdim($\mathcal{F}$) $\leq d+1$: We want to that for every $(x_1, \dots, x_{d+2})$, there exists a set of labels $(b_1, \dots, b_{d+2})$ that cannot be linearly separated.

(*Radon's theorem*): For any $(x_1, \dots, x_{d+2})$, there exists a partition of the points into sets S_1, S_2, s.t. convex hulls of S_1 and S_2 intersect.





How to use Radon's theorem: label pts in S_1, S_2 with +'s and –'s respectively.

Claim: no linear hyperplane perfectly separates S_1, S_2.

Pf: All pts in S_1 lie on one side of hyperplane, hence their convex hull does too.

All pts in S_2 lie on one side of hyperplane, hence their convex hull does too.

But, intersection is non-empty!

Bounding the VC dimension of a neural network

We will recursively use this to bound VC dimension of neural nets with binary activation function.

We will show: VC-dimension of **fully connected net** with **N** edges in total is O(**N log N**)

Main idea: growth function behaves nicely wrt to compositions and concatenations.

Bounding the VC dimension of a neural network

Claim 1: If $\mathcal{F} = \mathcal{F}_1 \times \mathcal{F}_2$ (**Cartesian product, i.e. concatenation**), we have $\Pi_{\mathcal{F}}(m) \leq \Pi_{\mathcal{F}_1}(m)\Pi_{\mathcal{F}_2}(m)$

Pf: Follows since any $\big(f(x_1), f(x_2), \dots, f(x_m)\big)$ can be written as

$$\Big(\big(f_1(x_1), f_2(x_1)\big), \big(f_1(x_2), f_2(x_2)\big), \dots, \big(f_1(x_m), f_2(x_m)\big)\Big).$$

Claim 2: If $\mathcal{F} = \mathcal{F}_1 \circ \mathcal{F}_2$ (**i.e. compositions**), we have $\Pi_{\mathcal{F}}(m) \leq \Pi_{\mathcal{F}_1}(m)\Pi_{\mathcal{F}_2}(m)$

Pf: $\left|\left\{\big(f(x_1), f(x_2), \dots, f(x_m)\big) \middle| f \in \mathcal{F}\right\}\right|$ can be written as

$$\left| \cup_{(y_1,\dots,y_m) \in \{(f_1(x_1), f_1(x_2), \dots, f_1(x_m)) |\, f_1 \in \mathcal{F}} \{(f_2(y_1), f_2(y_2), \dots, f_2(y_m))| f_2 \in \mathcal{F}_2\} \right|$$

$$\leq \sum_{(y_1,\dots,y_m) \in \{(f_1(x_1), f_1(x_2), \dots, f_1(x_m)) |\, f_1 \in \mathcal{F}} |\{(f_2(y_1), f_2(y_2), \dots, f_2(y_m))| f_2 \in \mathcal{F}_2\}|$$

$$\leq \Pi_{\mathcal{F}_1}(m)\Pi_{\mathcal{F}_2}(m)$$

Simple bounds on VC dimension of neural networks

We write a neural net as a sequence of concatenations and compositions.

Let $\mathcal{F}_{ij}$ be the set of functions (as we vary the input weights) calculated at the j^{th} node on the i^{th} layer.

If we view the **i^{th} layer** as a function, it can be written as the concatenation of the outputs of all the nodes, namely $\mathcal{F}_i = \mathcal{F}_{i1} \times \mathcal{F}_{i2} \times \cdots \times \mathcal{F}_{n_i}$

Furthermore, the **entire network** can be written as $\mathcal{F}_l \circ \mathcal{F}_{l-1} \circ \cdots \circ \mathcal{F}_1$

Using the lemmas on the previous slide, we have $\Pi_{\mathcal{F}}(m) \leq \Pi_{ij} \Pi_{\mathcal{F}_{ij}}(m)$

Note that each $\mathcal{F}_{ij}$ is a **hyperplane**, so by Sauer's lemma and the VC dimension bound we proved, we have $\Pi_{\mathcal{F}_{ij}}(m) \leq m^{d_{i-1}}$

Putting together, we have $\Pi_{\mathcal{F}}(m) \leq m^N$.(N is the # of params in net.)

If a size m set is shattered: we have $2^m \leq m^N \Rightarrow \mathrm{m} = \mathrm{O}(\mathrm{N} \log \mathrm{N})$

By definition of VC dimension, the proof follows.

"Rademacher" bounds

(Result by *Neyshabur-Bhojanapalli-Srebro '18*, following *Arora, Ge, Neyshabur, Zhang '18*)

Theorem: For a ReLU network with layers $W^1, W^2, \ldots, W^d$, at most h nodes per layer, output margin γ on a training set S, the generalization error can be bounded by

$$\tilde{O}\left(\frac{\sqrt{\mathrm{hd}^2 \max_{\mathrm{x}\in S} ||\mathrm{x}|| \underbrace{\Pi_{i=1}^{d} ||W^i||^2{}_2}_{\text{"Lipschitz constant"}} \underbrace{\Sigma_i^d \frac{||W^i||_F^2}{||W^i||_2^2}}_{\text{"Stable rank" (at most rank)}}}}{\gamma\sqrt{m}}\right)$$

“Rademacher” bounds

Result by *Bartlett-Foster-Telgarsky ’18* was based on covering number and Rademacher arguments.

Follow up by *Neyshabur-Bhojanapalli-Srebro ’18* was based on PAC-Bayes arguments.

We present an alternate proof by *Arora, Ge, Neyshabur, Zhang ’18* based on compression arguments.

Golowich-Rakhlin-Shamir ’19 sharpen the bounds using further Rademacher arguments.

Preliminaries

Some preliminaries:

A multiclass classifier f incurs can be associated with a margin loss

$$L_\gamma(f) = \mathbb{P}_{(x,y)}[f(x)[y] \leq \gamma + \max_{i \neq y} f(x)[i]]$$

Idea: "compress" neural network and count compressed networks.

Compressibility: Let $G_{\mathcal{A}} = \{g_A : A \in \mathcal{A}\}$ be a discrete set of classifiers. A classifier f is (γ, S)-compressible with respect to $G_{\mathcal{A}}$ if on a training set $\mathrm{S} = \{(x_i, y_i)\}$, we have:

$$\exists A \in \mathcal{A}, \qquad s.t.\ \forall x_i \in S, \qquad ||f(x_i) - g_A(x_i)||_\infty \leq \gamma$$

Generalization from compressibility

Idea: "compress" neural network and count compressed networks.

Compressibility: Let $G_{\mathcal{A}} = \{g_A : A \in \mathcal{A}\}$ be a discrete set of classifiers. A classifier f is (γ, S)-compressible with respect to $G_{\mathcal{A}}$ if on a training set $\mathrm{S} = \{(x_i, y_i)\}$, we have:

$$\exists A \in \mathcal{A}, \qquad s.t.\ \forall x_i \in S, \qquad ||f(x_i) - g_A(x_i)||_\infty \leq \gamma$$

Theorem: Suppose A is a set of q parameters each of which can have at most r discrete values and let $|S| = m$.

If the trained classifier f is (γ, S)- compressible with respect to $G_{\mathcal{A}}$, there exists an $A \in \mathcal{A}$, s.t. with high probability over S,

$$|L_0(g_A) - \hat{L}_\gamma(f)| \leq O\left(\sqrt{\frac{q \log r}{m}}\right)$$

Pf: Essentially Chernoff + union bound.

How do you compress?

How do you compress a neural network?

Lemma: For any matrix A, let $\hat{A}$ be the truncated version of A where the singular values of A smaller than $\delta||A||_2$ are removed. Then, $||\hat{A} - A||_2 \leq \delta||A||_2$ and $\hat{A}$ has rank at most $\frac{||A||_F^2}{\delta^2||A||_2^2}$.

Implied compression: write a rank r matrix $\hat{A}^{m \times n}$ as $\hat{U}^{m \times r}\hat{U}^{r \times n}$ which has (m+n)r parameters. (As opposed to mn.)

Proof: Denote by r the rank of $\hat{A}$. Max singular value of $\hat{A} - A$ is at most $\delta||A||_2$. Since the remaining singular values are at least $\delta||A||_2$, we have $||A||_F \geq ||\hat{A}||_F \geq \sqrt{r}\,\delta||A||_2$.

Perturbation after compression

Why spectral norm? Roughly captures "Lipschitzness" of network. Namely:

Lemma: Let $f_{\boldsymbol{w}}$ be a d-layer network with ReLU activations, and let $\boldsymbol{u} = \{U_i\}_{i=1}^{d}$ be perturbations, s.t. $||U_i||_2 \leq \frac{1}{d}||W_i||_2$. Then:

$$|f_{\boldsymbol{w}+\boldsymbol{u}}(x) - f_{\boldsymbol{w}}(x)| \leq e\, ||x||\, (\Pi_i ||W_i||_2) \sum_i \frac{||U_i||_2}{||W_i||_2}$$

Proof sketch: Induction on the error at layer i.

$$\begin{aligned}
|f^i_{\boldsymbol{w}+\boldsymbol{u}}(x) - f^i_{\boldsymbol{w}}(x)| &= |(W_i + U_i)\sigma(f^{i-1}_{\boldsymbol{w}+\boldsymbol{u}}(x)) - W_i\sigma(f^{i-1}_{\boldsymbol{w}}(x))| \\
&= |(W_i + U_i)(\sigma(f^{i-1}_{\boldsymbol{w}+\boldsymbol{u}}(x)) - \sigma(f^{i-1}_{\boldsymbol{w}}(x))) + U_i\,\sigma(f^{i-1}_{\boldsymbol{w}}(x))| \\
&\leq ||W_i + U_i||_2\, |f^{i-1}_{\boldsymbol{w}+\boldsymbol{u}}(x)) - f^{i-1}_{\boldsymbol{w}}(x)))| + ||U_i||_2\, |f^{i-1}_{\boldsymbol{w}}(x))|
\end{aligned}$$

Perturbation after compression

Why spectral norm? Roughly captures "Lipschitzness" of network. Namely:

Lemma: Let $f_{\boldsymbol{w}}$ be a d-layer network with ReLU activations, and let $\boldsymbol{u} = \{U_i\}_{i=1}^{d}$ be perturbations, s.t. $||U_i||_2 \leq \frac{1}{d}||W_i||_2$. Then:

$$|f_{\boldsymbol{w}+\boldsymbol{u}}(x) - f_{\boldsymbol{w}}(x)| \leq e\ ||x||\ (\Pi_i ||W_i||_2) \sum_i \frac{||U_i||_2}{||W_i||_2}$$

Proof sketch: Induction on the error at layer i.

$$|f^{i}_{\boldsymbol{w}+\boldsymbol{u}}(x) - f^{i}_{\boldsymbol{w}}(x)| \leq ||W_i + U_i||_2\ |f^{i-1}_{\boldsymbol{w}+\boldsymbol{u}}(x)) - f^{i-1}_{\boldsymbol{w}}(x)))| + ||U_i||_2\ |f^{i-1}_{\boldsymbol{w}}(x))|$$

$$\leq ||W_i + U_i||_2\ |f^{i-1}_{\boldsymbol{w}+\boldsymbol{u}}(x)) - f^{i-1}_{\boldsymbol{w}}(x)))| + ||U_i||_2 ||\mathrm{x}||\ \Pi_i ||W_i||_2$$

Using $||U_i||_2 \leq \frac{1}{d}||W_i||_2$, the lemma follows.

Putting things together

Lemma: Let $f_{\boldsymbol{w}}$ be a d-layer network with ReLU activations, and let $\boldsymbol{u} = \{U_i\}_{i=1}^{d}$ be perturbations, s.t. $||U_i||_2 \leq \frac{1}{d}||W_i||_2$. Then:

$$|f_{\boldsymbol{w}+\boldsymbol{u}}(x) - f_{\boldsymbol{w}}(x)| \leq e\ ||x||\ (\Pi_i ||W_i||_2) \sum_i \frac{||U_i||_2}{||W_i||_2}$$

Putting things together:

Applying compression lemma for $\delta = \frac{\gamma}{3d||x||\Pi_i||W^i||_2}$ gives $|f_{\boldsymbol{w}+\boldsymbol{u}}(x) - f_{\boldsymbol{w}}(x)| \leq \gamma$

Hence $|L_0(f_{\boldsymbol{w}+\boldsymbol{u}}) - L_\gamma(f_{\boldsymbol{w}})| \leq \gamma$

Putting things together

Lemma: For any matrix A, let $\hat{A}$ be the truncated version of A where the singular values of A smaller than $\delta||A||_2$ are removed. Then, $||\hat{A} - A||_2 \leq \delta||A||_2$ and $\hat{A}$ has rank at most $\frac{||A||_F^2}{\delta^2||A||_2^2}$.

Putting things together:

Applying compression lemma for $\delta = \frac{\gamma}{3d||x||\Pi_i||W^i||_2}$ gives $|f_{\boldsymbol{w}+\boldsymbol{u}}(x) - f_{\boldsymbol{w}}(x)| \leq \gamma$

Hence $|L_0(f_{\boldsymbol{w}+\boldsymbol{u}}) \ - L_\gamma(f_{\boldsymbol{w}})\ | \leq \gamma$

By truncation lemma, rank in layer i is at most $\frac{9d^2||x||\ ^2\Pi_i||W^i||_2^2}{\gamma^2}\frac{||W^i||_F^2}{||W^i||_2^2}$

Number of nodes is at most h per layer, so compressed encoding has num of params

$\frac{1}{\gamma^2}\mathrm{hd}^2\max_{\mathrm{x}\in S}||\mathrm{x}||\ \Pi_{i=1}^d||W^i||^2{}_2\sum_i^d\frac{||W^i||_F^2}{||W^i||_2^2}$. Applying compression lemma, proof follows.