On the Optimal Memorization Power of ReLU Neural Networks

ICLR 2022

Gal Vardi, Gilad Yehudai and Ohad Shamir

Weizmann Institute

Memorization

The problem

What is the minimal size $s(n)$ of NN that suffices to interpolate every size-n dataset?

size = number of neurons or parameters

Memorization

The problem

What is the minimal size $s(n)$ of NN that suffices to interpolate every size-n dataset?

size = number of neurons or parameters

- A natural notion of expressiveness.
- Related to the *double descent* phenomenon: the second descent starts after the *interpolation threshold*.

Memorization – known results

Studied since the 80's...

Some results for ReLU networks:

- Depth 2: $4 \cdot \lceil n/d \rceil$ neurons, for n points in general position in $\mathbb{R}^d$ [Bubeck et al. 2020, Baum 1988].
- Depth 3: $O(\sqrt{n})$ neurons, $O(n)$ parameters [Yun et al. 2019].
- Deep: $O\left(n^{2/3} + \log(1/\delta)\right)$ parameters for δ-separated data [Park et al. 2021].

Memorization – known results

Studied since the 80's...

Some results for ReLU networks:

- Depth 2: $4 \cdot \lceil n/d \rceil$ neurons, for n points in general position in $\mathbb{R}^d$ [Bubeck et al. 2020, Baum 1988].
- Depth 3: $O(\sqrt{n})$ neurons, $O(n)$ parameters [Yun et al. 2019].
- Deep: $O\left(n^{2/3} + \log(1/\delta)\right)$ parameters for δ-separated data [Park et al. 2021].

Lower bound (from VCdim):

$\Omega(\sqrt{n})$ parameters [Goldberg & Jerrum 1995]

Optimal memorization capacity

Theorem

Let $(\mathrm{x}_1, y_1), \ldots, (\mathrm{x}_n, y_n) \in \mathbb{R}^d \times \{1, \ldots, C\}$ *where* d *is constant,* $\|\mathrm{x}_i\| \leq r$ *for every* i*, and* $\|\mathrm{x}_i - \mathrm{x}_j\| \geq \delta$ *for every* $i \neq j$*. Then, there exists a ReLU network* $F : \mathbb{R}^d \to \mathbb{R}$ *with* $\tilde{O}\left(\sqrt{n}\right)$ *parameters, such that* $F(\mathrm{x}_i) = y_i$ *for every* $i \in [n]$*.*

$\tilde{O}(\cdot)$ *hides log factors in* n, C, r, δ^{-1}

Optimal memorization capacity

Theorem

Let $(\mathrm{x}_1, y_1), \ldots, (\mathrm{x}_n, y_n) \in \mathbb{R}^d \times \{1, \ldots, C\}$ *where* d *is constant,* $\|\mathrm{x}_i\| \leq r$ *for every* i*, and* $\|\mathrm{x}_i - \mathrm{x}_j\| \geq \delta$ *for every* $i \neq j$*. Then, there exists a ReLU network* $F : \mathbb{R}^d \to \mathbb{R}$ *with* $\tilde{O}\left(\sqrt{n}\right)$ *parameters, such that* $F(\mathrm{x}_i) = y_i$ *for every* $i \in [n]$*.*

$\tilde{O}(\cdot)$ *hides log factors in* n, C, r, δ^{-1}

Matches the $\Omega(\sqrt{n})$ lower bound (up to log factors)

$\Rightarrow$ Memorizing all size-n datasets is not harder than shattering a single size-n set (up to log factors...)

Is depth required for efficient memorization?

- Our construction: $\tilde{O}(\sqrt{n})$ depth. Can we do better?
- A lower bound implied by [Bartlett et al. 2019]:
  - Memorizing n points with depth L requires $\tilde{\Omega}(n/L)$ parameters.

Is depth required for efficient memorization?

- Our construction: $\tilde{O}(\sqrt{n})$ depth. Can we do better?
- A lower bound implied by [Bartlett et al. 2019]:
  - Memorizing n points with depth L requires $\tilde{\Omega}(n/L)$ parameters.

Theorem

Let $1 \leq L \leq \sqrt{n}$. We can memorize n points with depth L and $\tilde{O}(n/L)$ parameters.