On the Complexity of Learning Sparse Functions with Statistical and Gradient Queries

NeurIPS 2024

Nirmit Joshi
TTIC

Theodor Misiakiewicz
TTIC—>Yale

Nati Srebro
TTIC

Complexity of learning with gradient based algorithms on NNs

There has been a lot of interest in recent years in investigating the complexity of learning with neural networks. Which functio easy? Which ones are difficult to learn? etc.

[Abbe et al. 23,24] [Glassgow 24] [Edleman et al. 19,22] & many more

Learning Sparse Function

A junta problem with P "relevant coordinates" out of $d \gg P$ total coordinates of the input $\boldsymbol{x} \in \mathcal{X}^d$ corresponds to learning a family of distributions; $\mathcal{H}_\mu^d := \left\{ \mathcal{D}_{\mu,s}^d : s \text{ is a non-repeating sequence from } [P] \to [d] \right\}$ where $\mathcal{D}_{\mu,s}^d$ is supported on $\mathcal{Y} \times \mathcal{X}^d$ such that $\mu := (\mu_x, \mu_{y|z})$ specifies marginal and link function respectively as below

$$\boldsymbol{x} \sim \mu_x^d \quad \text{and} \quad y \mid (x_{s(1)}, \ldots, x_{s(P)}) \sim \mu_{y|z} \quad \boldsymbol{z} = (x_{s(1)}, \ldots, x_{s(P)}) \text{ is the "support"}$$

Question: What is the complexity of Learning a specific problem μ, especially using (S)GD on NNs?

e.g. linear functions are learned in $O(d)$ time but parities take $\Omega(d^P)$.

Motivation

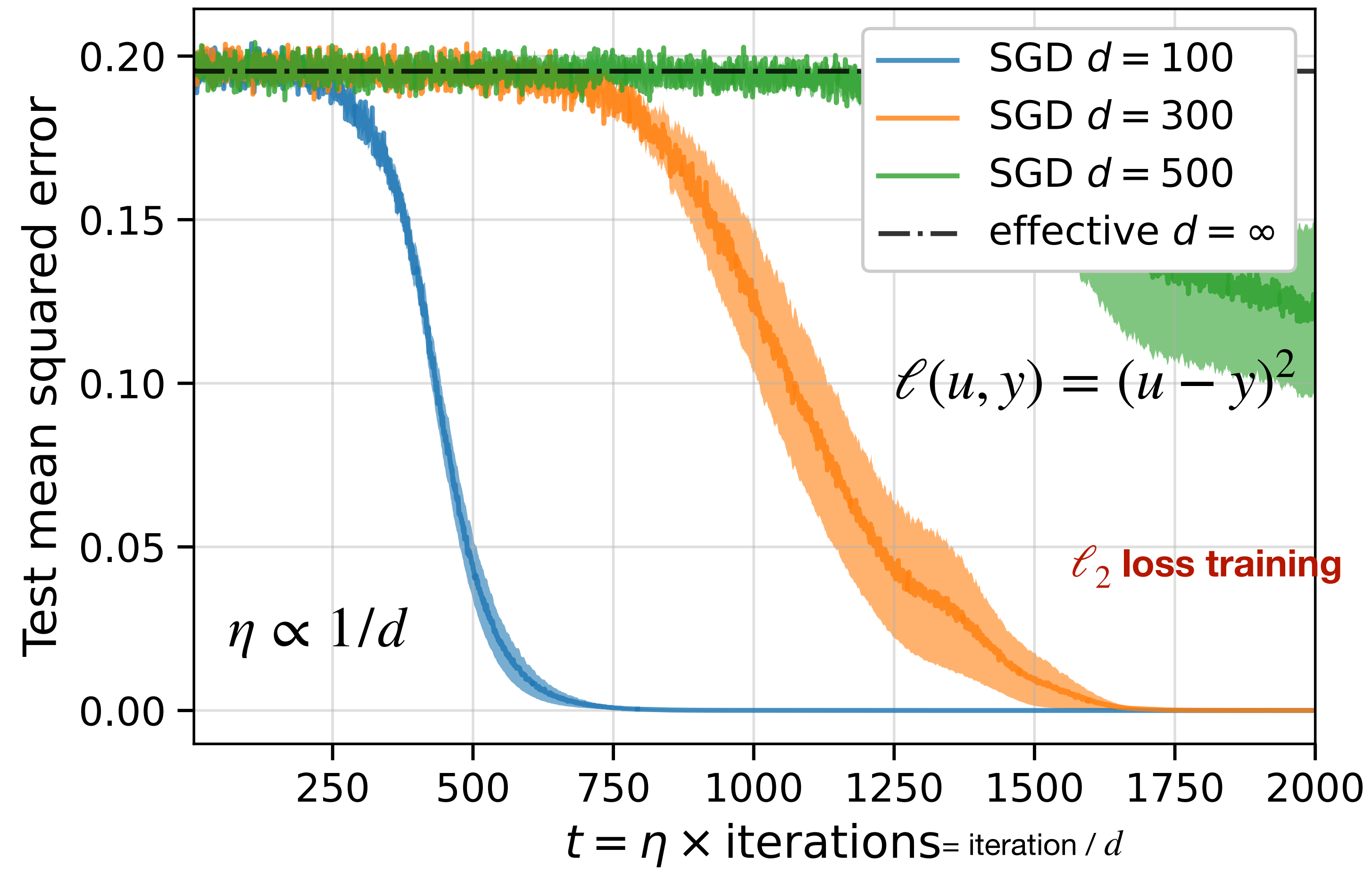
- A popular approach has been to show Correlation Statistical Query (CSQ) lower bound which captures learning with gradient queries

- For e.g [Abbe et al. 22,23] consider the special case of $\mu^d \equiv \mathrm{Unif}(\{-1,+1\}^d)$ unveiling rich hierarchical structure "leap complexity" and show CSQ lower bounds. The complexity grows as $\Omega(d^{\mathsf{Leap}(\mu)})$

Gradient Queries:

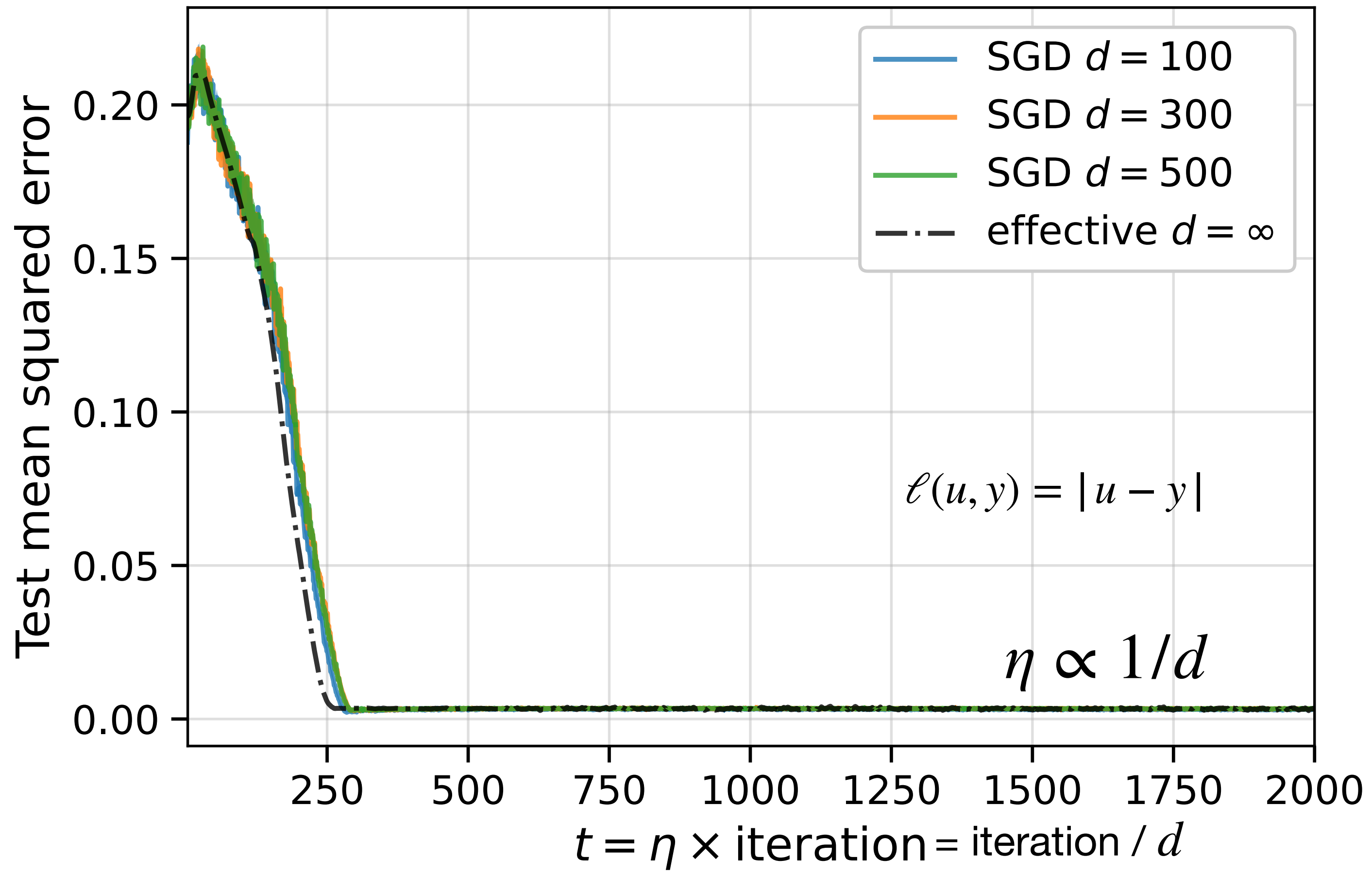
$$\mathbb{E}\left[\nabla_\theta\left(y - f_\theta(x)\right)^2\right] = -2\mathbb{E}\left[y\,\nabla_\theta f_\theta(x)\right] + 2\mathbb{E}\left[f_\theta(x)\,\nabla_\theta f_\theta(x)\right]$$

Correlation Statistical Query



Main Observation: The complexity of learning $\mu_{y|z} = h_*(z)$ with online SGD changes when we change the loss....

The goals of this work is to characterize the loss-specific complexity and in much greater generality beyond boolean hypercube.



CSQ lower bound is escaped. Why?
On changing the loss, the gradient queries $\nabla_\theta \ell(f_\theta(x), y)$ are more powerful than correlation queries

(C)SQ and Differentiable Learning Queries (DLQ)

- A $\mathcal{Q}$-restricted SQ learner with tolerance τ issues a query $\phi \in \mathcal{Q} \subseteq L^2(\mathcal{Y} \times \mathcal{X}^d)$ which is $\phi : \mathcal{Y} \times \mathcal{X}^d \to \mathbb{R}$ (with controlled scale) and receives a response v such that

$$|v - \mathbb{E}_{\mathcal{D}}[\phi(y, \boldsymbol{x})]| \leq \tau$$

1. $\mathcal{Q}_{\mathsf{SQ}} = L^2(\mathcal{Y} \times \mathcal{X}^d)$ (with scale controlled)

2. $\mathcal{Q}_{\mathsf{CSQ}} \subset \mathcal{Q}_{\mathsf{SQ}}$ contains $\phi(y, \boldsymbol{x}) = y \cdot \tilde{\phi}(\boldsymbol{x})$

3. $\mathcal{Q}_{\mathsf{DLQ}_\ell} \subset \mathcal{Q}_{\mathsf{SQ}}$ contains of $\phi(y, \boldsymbol{x})$ of the form

$$\phi(y, \boldsymbol{x}) = \frac{\partial}{\partial \omega} \ell(y, f(\boldsymbol{x}, \omega)) \,|_{\omega=0} \; ; \quad f : \mathcal{X}^d \times \mathbb{R} \to \mathbb{R}$$

Differentiable Learning Queries with loss ℓ

Main Result: Characterizing the Complexity of SQ, CSQ & DLQ

Adaptive Query Complexity

Any "***adaptive***" learner $\mathsf{A} \in \{\mathsf{SQ}, \mathsf{CSQ}, \mathsf{DLQ}_\ell\}$, with precision τ requires q queries s.t.choosing

$$q/\tau^2 = \Omega(d^{\mathsf{Leap}_\mathsf{A}(\mu)})$$

There exists a learner with with $q/\tau^2 = O(d^{\mathsf{Leap}_\mathsf{A}(\mu)})$.

$$\mathsf{Leap}_\mathsf{A}(\mu) = \min_{\substack{U_1,\ldots,U_r \in \mathscr{C}_\mathsf{A} \\ \cup_{i\in[r]} U_i = [P]}} \max_{i\in[r]} \left| U_i \setminus \cup_{j=1}^{i-1} U_j \right|$$

System of Detectable Subsets

A set $U \in \mathscr{C}_\mathsf{A}$, is detectable by the method A, if there exists $T(y) \in \Psi_\mathsf{A}$ ("the test functions set")

And zero-mean functions T_i (i.e. $\mathbb{E}_{z_i\sim\mu_x}[T_i(z_i)] = 0,\ \forall i \in U$,

$$\text{s.t. } \mathbb{E}_{z,y\sim\mu_{y|z}}\left[T(y)\prod_{i\in U} T_i(z_i)\right] \neq 0$$

Method Specific Test Functions Set Ψ (Query Model Specific) :

(1) $\Psi_{\mathsf{SQ}} = L^2(\mu_y)$ (all squre integrable functions)

(2) $\Psi_{\mathsf{CSQ}} = \{y \mapsto y\}$ (just identity)

(3) $\Psi_{\mathsf{DLQ}_\ell} = \{y \mapsto \partial_1 \ell(u, y), u \in \mathbb{R}\}$ (i.e. gradient w.r.t first argument)

Non-Adaptive Query Complexity

Any "***non-adaptive***" learner $\mathsf{A} \in \{\mathsf{SQ}, \mathsf{CSQ}, \mathsf{DLQ}_\ell\}$, with precision τ requires q queries s.t.choosing

$$q/\tau^2 = \Omega(d^{\mathsf{Cover}_\mathsf{A}(\mu)})$$

There exists a learner with with $q/\tau^2 = O(d^{\mathsf{Cover}_\mathsf{A}(\mu)})$.

$$\mathsf{Cover}_\mathsf{A}(\mu) = \max_{i\in[P]} \min_{i\in U, U\in\mathscr{C}_\mathsf{A}} \left| U \right|$$

Other Results and Connection with SGD on NNs

Relationship between SQ, CSQ, DLQ

For classification $\mathcal{Y} = \{-1, +1\}$ (like parities): SQ and CSQ complexities are equal.

$$\mathscr{C}_{\text{SQ}} = \mathscr{C}_{\text{CSQ}};\ \text{Leap}_{\text{SQ}} = \text{Leap}_{\text{CSQ}};\ \text{Cover}_{\text{SQ}} = \text{Cover}_{\text{CSQ}}$$

For regression, there can be arbitrary separation.

There exists a problem μ such that
$\text{Leap}_{\text{SQ}}(\mu) = 1$ but $\text{Leap}_{\text{CSQ}}(\mu) = P - 1$

Finally, for the squared loss, we have $\mathscr{C}_{\text{DLQ}_{\ell:(u,y)\mapsto(u-y)^2}} = \mathscr{C}_{\text{CSQ}}$.

But.., for the absolute loss, we have $\mathscr{C}_{\text{DLQ}_{\ell:(u,y)\mapsto|u-y|}} = \mathscr{C}_{\text{SQ}}$.

ℓ_1 loss is "universal" e.g. always learns at SQ complexity

$q/\tau^2 = \Theta(d^{k_*})$	**Adaptive**	**Non-Adaptive**
SQ	$k_* = \text{Leap}_{\text{SQ}}(\mu)$	$k_* = \text{Cover}_{\text{SQ}}(\mu)$
CSQ	$k_* = \text{Leap}_{\text{CSQ}}(\mu)$	$k_* = \text{Cover}_{\text{CSQ}}(\mu)$
DLQ_ℓ	$k_* = \text{Leap}_{\text{DLQ}}(\mu)$	$k_* = \text{Cover}_{\text{DLQ}}(\mu)$

Stochastic Gradient Descent on Neural Network:

On Hypercube $\text{Leap}_{\text{DLQ}_\ell} = 1$ sharply characterizes what problems are learnable in $O(d)$ scaling with online SGD with a loss ℓ.

Leap=1 is a.k.a. merged staircase property for ℓ_2 loss [Abbe et al 2022]

- Online SGD with loss ℓ strongly learns junta problems $\text{Leap}_{\text{DLQ}_\ell} = 1$ in $O(d)$ samples/iterations.
- If $\text{Leap}_{\text{DLQ}_\ell} > 1$, the dynamics get stuck in suboptimal saddle in $O(d)$ iterations.

Do check out the paper!
See you at the poster session!