NEURAL INFORMATION PROCESSING SYSTEMS

Can Neural Networks Achieve Optimal Computational-Statistical Trade-off?

An Analysis on Single-index Model

Siyu Chen (Yale) Beining Wu (UChicago) Miao Lu (Stanford)

Zhuoran Yang (Yale) Tianhao Wang (TTIC)

NeurIPS 2024 Workshop on Mathematics of Modern Machine Learning

Introduction: Learning Single-Index Models

Gaussian single-index model $\mathbb{P}_{\boldsymbol{\theta}^\star}: \quad y \sim p(\,\cdot \mid \langle \boldsymbol{\theta}^\star, z\rangle), \quad z \sim \mathcal{N}(0, I_d)$

- **Goal:** learning the unknown signal $\boldsymbol{\theta}^\star \in \mathbb{R}^d$, not knowing $p(\,\cdot \mid \cdot\,)$
- **Assume:** $p(\,\cdot \mid \cdot\,)$ has generative exponent $s^\star \in \mathbb{N}$ (Damian et al, 24)

$$\mathbb{P}_y \otimes \mathbb{P}_z \longleftarrow \frac{\mathbb{P}_{\theta^\star}(y,z)}{\boxed{\mathbb{Q}}(y,z)} \overset{L^2(\mathbb{Q})}{=} 1 + \sum_{s \geq \boxed{s^\star}}^{\infty} \zeta_s(y) \cdot \mathrm{He}_s\big(\langle \theta^\star, z\rangle\big) \longrightarrow \text{Generative Exponent}$$

Information Theoretic Limit

Theorem (Bach 17, …)
Information theoretically, $n = \Omega(d)$ is necessary and sufficient to recover the true signal $\boldsymbol{\theta}^\star \in \mathbb{R}^d$.

- Generally, this requires exponential computing to recover the signal.
- What's the statistical complexity of gradient-based NN learner?

Baseline: Learning with Information Exponent $k^\star$

$$p(x) = \sum_{i=0}^{\infty} \alpha_i \cdot \mathrm{He}_i(x), \quad k^\star = \min\{k \in \mathbb{N}_+ : \alpha_i \neq 0\}\,.$$

Theorem For polynomial link function with information exponent $k^\star \in \mathbb{N}$

- (Arous et al. 21) two-layer NN trained by (variants) of GD can learn $\theta^\star$ using $n = \Omega(d^{\Theta(k^\star)})$ number of samples.

- (Damian et al. 23) two-layer NN trained by GD with landscape smoothing can learn with $n = \Omega(d^{k^\star/2})$

- Computational-statistical gap exists for $k^\star > 2$.
- Inevitable under Correlational Statistical Query (CSQ) framework.
- Does CSQ framework characterize the fundamental stat limit of all gradient-based algorithms ?

CSQ vs SQ Framework

CSQ learner: algorithm accesses noisy queries of $y \cdot \phi(z)$:

$$\left| \tilde{q} - \mathbb{E}_{y,z}[y \cdot \phi(z)] \right| \leq \tau$$

correlational

- Does CSQ framework characterize the fundamental stat limit of all gradient-based algorithms ?

SQ learner: algorithm accesses noisy queries of $\phi(y, z)$:

$$\left| \tilde{q} - \mathbb{E}_{y,z}[\phi(y, z)] \right| \leq \tau$$

Leverage higher order information in gradient !

- **Example:** batch-reusing for polynomial link function, $n = \widetilde{O}(d)$ (Dandi et al. 24, Damian et al. 24).
- No computational-statistical gap (up to log) for polynomial link.

SQ Lower Bound and Prior Arts

SQ learner: algorithm accesses noisy queries of $\phi(y, z)$:

$$\left| \widetilde{q} - \mathbb{E}_{y,z}[\phi(y, z)] \right| \leq \tau$$

Theorem (SQ lower bound; Damian et al. 24)

Under SQ, for link function with generative exponent $s^\star$, to learn $\boldsymbol{\theta}^\star$ using polynomial compute, it requires $n = \Omega(d^{s^\star/2})$ samples.

- **Prior arts:** polynomial link function ($s^\star \leq 2$).
- **This work: can we achieve SQ lower bound by gradient-based algorithms for general link function with arbitrary $s^\star$?**

Failure of Vanilla SGD under Square Loss

$f(z;\theta,a) = a \cdot \sigma(\langle \theta, z \rangle)$

- Illustration: the rescaled gradient (single data, single neuron):

$$g = (2a)^{-1} \cdot \nabla_{\boldsymbol{\theta}} \big(f(z;\boldsymbol{\theta},a) - y\big)^2 \approx y \cdot \sigma'(\langle z, \boldsymbol{\theta} \rangle) \cdot z$$

- Moment calculation of the gradient:

$$\mathbb{E}_{\mathbb{P}_{\theta^\star}}[g] \approx \underbrace{\mathbb{E}_{\mathbb{Q}}\left[y\right] \cdot \mathbb{E}_{\mathbb{Q}}\left[\sigma'(\langle z, \boldsymbol{\theta} \rangle) \cdot z\right]}_{\text{direct bias}} + \sum_{s \geq s^\star} \underbrace{\mathbb{E}_{\mathbb{Q}}\left[y \cdot \zeta_s(y)\right] \cdot \mathbb{E}_{\mathbb{Q}}\left[\mathrm{He}_s(\langle \boldsymbol{\theta}^\star, z \rangle) \cdot \sigma'(\langle z, \boldsymbol{\theta} \rangle) \cdot z\right]}_{\text{informative queries}}$$

Challenge 1: Zero correlation
$\mathbb{E}_{\mathbb{Q}}[y \cdot \zeta_{s^\star}(y)] = 0$

Challenge 3: Non-polynomial

- SNR - Squared alignment of the normalized gradient:

$$\langle \mathbb{E}_{\mathbb{P}_{\theta^\star}}[g], \theta^\star \rangle^2 / \mathbb{E}_{\mathbb{P}_{\theta^\star}}\left[\|g]\|_2^2\right] \simeq d^{-s^\star}$$

Non-trivial alignment requires:

Challenge 2: Low SNR

$$n \cdot \mathrm{SNR}_{1-\text{sample}} \gg d^{-1} \Leftrightarrow n = \Omega(d^{s^\star - 1})$$

Algorithm Overview

- Questions: Can we devise proper algorithm that tackles these issues?

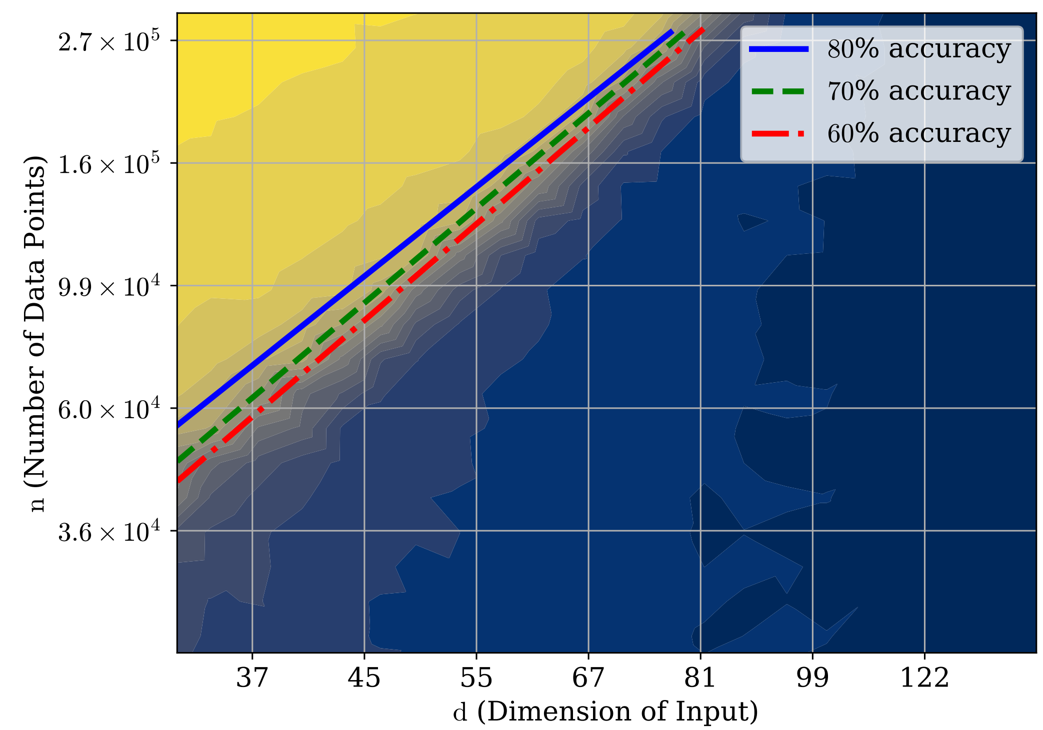
Architecture: 2-layer neural network

high-pass activation

$$f(z;\theta,a) = \sum_{m=1}^{M} \boxed{a_m} \cdot \boxed{\sigma}(\langle \theta_m, z \rangle)$$

fixed and small

$p(x) = x^2 \cdot \exp(-x^2), \quad s^\star = 4$



Algorithm: online batched SGD with

$$\widetilde{\boldsymbol{\theta}}_m^{(t+1)} = \boldsymbol{\theta}_m^{(t)} + \eta \bar{g}_m^{(t)}, \quad \boldsymbol{\theta}_m^{(t+1)} = \widetilde{\boldsymbol{\theta}}_m^{(t+1)} / \|\widetilde{\boldsymbol{\theta}}_m^{(t+1)}\|_2$$

Gradient direction $\bar{g}_m^{(t)}$ computed using:

- General gradient oracle to perform label transformation (challenge 1)
- Weight perturbation to explore loss landscape (challenge 2&3)

$$\tilde{g}(w; y, z) = \psi(y, \langle w, z \rangle) \cdot z$$

$$w_{m,l}^{(t)} = \frac{\gamma \boldsymbol{\theta}_m^{(t)} + \xi_{m,l}^{(t)}}{\|\gamma \boldsymbol{\theta}_m^{(t)} + \xi_{m,l}^{(t)}\|_2}, \quad \xi_{m,l}^{(t)} \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}(\mathbb{S}^{d-1})$$

$$\bar{g}_m^{(t)} = \frac{1}{nL} \sum_{i=1}^{n} \sum_{l=1}^{L} \tilde{g}(w_{m,l}^{(t)}, y_i^{(t)}, z_i^{(t)}) + \text{debias}$$

Statistical Complexity: Matching SQ Lower Bound

Assumption 1 (Gradient Oracle)

- Grad oracle has polynomial-like tails
- Grad oracle is high-pass: (i) $\widehat{\psi}_s(y) = 0$ for $s \leq s^\star - 2$; (ii) it holds that $|\mathbb{E}_{\mathbb{P}}[\zeta_{s^\star}(y)\,\widehat{\psi}_{s^\star-1}(y)]| \geq C$

Theorem (Chen et al. 24)

With Assumption 1, let mini-batch size $n = \widetilde{\Theta}(d^{s^\star/2})$, perturbation $\gamma = d^{-1/4}$, and neuron replica number $L = \widetilde{\Theta}(d^{(s^\star+1)/2})$, running online batched SGD with LR $\eta \geq 2$ for $T = \Theta(\log d)$ steps, we have at least $\Omega(M)$ neurons satisfy $|\langle \boldsymbol{\theta}_m^{(T)}, \boldsymbol{\theta}^\star \rangle| \geq 1 - O(d^{-\epsilon})$

- Instances:
  - batch-reusing: $\psi(y, x) = y\sigma'(x) + y\sigma'(x + y\sigma'(x))$;
  - modified loss: $\psi(y, x) = \partial_f \ell(y, 0) \cdot \sigma'(x)$.

Sparse Prior: New SQ Lower Bound and Matching Upper Bound

Sparse signal prior:

$\phi^\star$ is a uniformly sampled random k-subset of $[d]$;

$\theta^\star \mid \phi^\star \sim \mathrm{Uniform}(\mathbb{S}^{k-1}(\phi^\star))$.

Theorem (Sparse, Chen et al. 24)
Suppose that $k = o(\sqrt{d})$. Let $n = \widetilde{\Theta}(k^{s^\star})$, $\gamma = k^{-1/2}$ and $L = \widetilde{\Theta}(k^{(s^\star+3)/2})$. Running online batched SGD with projection onto the top-k support and learning rate $\eta > 2$ for constant steps, we have $\Omega(M)$ neurons with $|\langle \boldsymbol{\theta}_m^{(T)}, \boldsymbol{\theta}^\star \rangle| \geq 1 - O(d^{-\epsilon})$

Theorem (SQ Lower bound, Chen et al. 24) Any SQ type algorithm requires sample size $n = \Omega(k^{s^\star})$ to obtain nontrivial alignment.

Thanks for Attending!