Learning Low-Rank Polynomials with Neural Networks

Jason D. Lee, Minshuo Chen, Yu Bai, Tuo Zhao, Huan Wang, Caiming Xiong, and Richard Socher.

Princeton University
Subjective statements are all due to JDL. Theorems are joint work.
Many missing references

September 4, 2020

One of the Most "Natural" Questions in DL Theory

Two-layer Teacher Network.

$$f^*(x) = \sum_{j=1}^{r} a_j^* \sigma(w_j^{*\top} x).$$

Problem

Given data from a two-layer teacher, learn to accuracy

$$\mathbb{E}(f^*(x) - f(x))^2 < \epsilon.$$

What is a good result?

Potential results by strength

- Information-theoretic sample complexity is $n \asymp dr$, which is attainable in well-specified case.
- Very good result if computationally efficient: $n \asymp \mathrm{poly}(d, r)$.
- If σ is monomial/polynomial, polynomial kernel will attain $n \asymp d^{\mathsf{deg}(\sigma)} r^2$.

Lower bounds

We shouldn't stop trying because of lower bounds, but we should know what they say.

Lower bounds for discrete distribution (Adam Klivans).

These results rule out any algorithm that does not utilize distribution assumptions/assumptions on W^* that learn in $\mathrm{poly}(d, r)$ (probably even $d^{o(r)}$ is hard).

- Learning intersection of halfspaces (Klivans& Sherstov, Livni et al.)
- Decision trees
- Juntas

Lower bounds II and "breaking" lower bounds

Distribution-specific lower bounds

Even if $p(x)$ is isotropic Gaussian, there are some recent negative results (SQ lower bounds):

- For ReLU teacher network, need d^r queries (Diakonikolas et al., similar result by Goel et al.)

Lower bounds II and "breaking" lower bounds

Distribution-specific lower bounds

Even if $p(x)$ is isotropic Gaussian, there are some recent negative results (SQ lower bounds):

- For ReLU teacher network, need d^r queries (Diakonikolas et al., similar result by Goel et al.)

We should not be discouraged by lower bounds as long as we have algorithmic ideas:

Algorithms (well-specified case, parameter recovery)

- Spectral Methods (Janzamin et al., Zhong et al.): Learn teacher networks if $r \leq d$ and $\sigma_{\min}(W^*) > 0$.
- Tensor method in disguise (GLM, Li et al.): SGD can estimate parameters but under even stronger assumptions.

Functional Analysis Viewpoint

Function spaces

For $f^*(x) = \sum_{j=1}^m a_j \sigma(w_j^\top x)$, write it as

$$f(x) = \int \rho(w)\sigma(w^\top x)dw = \rho^\top \phi(x),$$

where $\phi(x)[w] = \sigma(w^\top x)$ with index set $w \in S^{d-1}$.
Two "natural" function spaces:

- $F_2(B) = \{f : f(x) = \rho^\top \phi(x), \|\rho\|_2^2 \leq B^2\}$ is an ℓ_2 space (RKHS, Rahimi-Recht,Cho and Saul)
- $F_1(B) = \{f : f(x) = \rho^\top \phi(x), \|\rho\|_1 \leq B\}$ is an ℓ_1 sparsity-type space known as convex neural net (Banach, Bengio et al., Bach, ...)

Summary of what is known for two spaces.

F_1

- The global minimum of ℓ_2-norm on all parameters is F_1.
- Mean field aims to learn all of F_1, so does implicit regularization (Chizat-Bach, Nacson et al., Lyu et al., Wei et al.)
- F_1 **adapts to low-dimensional structure**.
  1. $n \asymp d\|f^*\|_{F_1}^2/\epsilon^2$
  2. If $f^*(x) = p(Ux)$, for $U = r \times d$ and $p(\cdot)$ is degree q polynomial, then $n_{F_1}(p(Ux)) = dr^{2q}$.
  3. Learn width r teacher networks in complexity $n_{F_1}(\text{teacher net}) = d \cdot \text{poly}(r)$.
- F_1 almost certainly **cannot be efficiently learned, since it includes two-layer teacher networks of width** r. All the previously mentioned lower bounds apply to F_1.
- Opinion: Unlikely that mean field or any SGD approach will yield polynomial-time learning of F_1.

Summary of what is known for two spaces.

F_2

- Computationally efficient via SGD.
- Does not adapt to low-dimensional structure. If $f^*(x) = p(Ux)$, then $n_{F_2} \asymp d^q$ even if $U = 1 \times d$.
- Teacher network has $\gtrsim e^d$ (probably infinite) F_2 norm (Yehudai and Shamir) and sample complexity $n_{F_2} \gtrsim d^{\mathrm{poly}(1/\epsilon)}$.
- Essentially the same as NTK as far as theoretical results bounds go.

Finding the sweet spot.

Goal.

The goal is to find an in-between space that

- Adapts to low-dimensional structure
- Computationally tractable via SGD.

Finding the sweet spot.

Goal.

The goal is to find an in-between space that

- Adapts to low-dimensional structure
- Computationally tractable via SGD.

Obvious guess is ℓ_p for $1 < p < 2$, but I don't think this leads to tractable algorithms.

Motivation

Monomial Activation

For quadratic activation (and monomial activation),

$$\int \rho(w)(w^\top x)^2 = \langle \int \rho(w) w w^\top, x x^\top \rangle.$$

- F_2 is a frobenius norm inductive bias.
- F_1 is a nuclear norm inductive bias.

Motivation

Monomial Activation

For quadratic activation (and monomial activation),

$$\int \rho(w)(w^\top x)^2 = \langle \int \rho(w) w w^\top, x x^\top \rangle.$$

- F_2 is a frobenius norm inductive bias.
- F_1 is a nuclear norm inductive bias.

- This suggests rank as a measure of "low-dimensional" latent structure.
- For monomial activation, this corresponds to the width of the teacher network.

Low-rank Polynomials

Definition

(Low rank polynomial) f^* is a rank r polynomial of degree p if

$$f^*(x) = \sum_{s=1}^{r} a_s^* (w_s^{*\top} x)^{p_s},$$

where $|a_s^*| \leq 1$, $\mathbb{E}[(w_s^{*\top} x)^{2p_s}] \leq 1$, and $p_s \leq p$.

Low-rank Polynomials

Definition

(Low rank polynomial) f^* is a rank r polynomial of degree p if

$$f^*(x) = \sum_{s=1}^{r} a_s^* (w_s^{*\top} x)^{p_s},$$

where $|a_s^*| \leq 1$, $\mathbb{E}[(w_s^{*\top} x)^{2p_s}] \leq 1$, and $p_s \leq p$.

- Teacher networks with polynomial activation of bounded degree and analytic activation (approximately).
- Constant depth teacher networks with polynomial activation.
- Real reason: I will show you a non-trivial learning guarantee with SGD.

Baseline approaches

- Using F_2 to learn this class needs $\gtrsim d^p$ samples.
- Using F_1 to learn needs at most $d \cdot \mathrm{poly}(r)$ samples (nearly information-theoretic optimal).
- SGD+Signed Dropout needs $d^{p-1} \cdot \mathrm{poly}(r, p)$ samples (via the Quadratic NTK proof technique in Bai and Lee).

Baseline approaches

- Using F_2 to learn this class needs $\gtrsim d^p$ samples.
- Using F_1 to learn needs at most $d \cdot \mathrm{poly}(r)$ samples (nearly information-theoretic optimal).
- SGD+Signed Dropout needs $d^{p-1} \cdot \mathrm{poly}(r, p)$ samples (via the Quadratic NTK proof technique in Bai and Lee).

Still a very large gap between $d \cdot \mathrm{poly}(r)$ and $d^{p-1} \cdot \mathrm{poly}(r, p)$.

Main Theorem

Theorem

SGD+Signed Dropout on a three-layer neural net architecture (polynomial width) learns with $n \asymp d^{p/2} \cdot \frac{\mathrm{poly}(r,p)}{\epsilon^4}$ *in time* $n \cdot \mathrm{poly}(d, r, p, \frac{1}{\epsilon})$.

Assumption:

- Moment assumptions on x. Any multivariate Gaussian or elliptical distribution on sphere is fine.

How to attain this.

Architecture

3-layer network:

$$f(x) = \sum_{j=1}^{m} a_r \sigma(w_r^\top g(x))$$
$$g(x)_l = \sigma(v_l^\top x + b).$$

- We will only train w_r. The a_r, v_l are randomly initialized and fixed.
- It is crucial to have a 3-layer architecture, our results are not attainable with only a two-layer network (lower bound).

Alg: SGD with data-dependent regularizer

Step 1: Estimate covariance $\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} g(x_i)g(x_i)^\top$
Step 2: Run SGD +Signed Dropout (AzLL, Bai and Lee) on

$$L(w_1, \ldots, w_m) = \frac{1}{n}\sum_{i=1}^{n} \ell(f_W(x_i), y_i) + \lambda \|W\hat{\Sigma}^{1/2}\|_{2,4}^4$$

- Signed Dropout: Modify the gradient per neuron $z_r \nabla_{w_r} L(W)$ for z_r Rademacher.
- Prevents the linearized model from memorizing (NTK).
- Allows for learning rate $O(m^{0.25})$ larger than NTK.

Why it works.

Proof sketch (assuming I know the input is Gaussian and $f(x) = (\beta^\top x)^p$):

1. Let $g(x)_l$ be Hermite polynomial basis of degree $\frac{p}{2}$ for $1 \leq l \leq D := d^{p/2}$.
2. Via the hermite, we can express $(\beta^\top x)^{p/2} = \theta^\top g(x)$.
3. The second layer input is the hermite polynomials. It needs to learn $f(x) = (\theta^\top g(x))^2$.
4. QNTK is very good at learning quadratic functions of the input

$$(\theta^\top g(x))^2 = \sum_{j=1}^{m} \sigma''(w_{0,j}^\top g(x))(\theta^\top g(x))^2.$$

Potential Improvements

Probably within reach.

Train the first layer v_r's and show this can improve ϵ dependence. Hope:

$$n \asymp d^{p/2} \cdot \frac{\text{poly}(r,p)}{\epsilon^4} \to n \asymp d^{p/2} \cdot \frac{\text{poly}(r,p)}{\epsilon^2}.$$

- Currently, $\frac{1}{\epsilon^2}$ is due to the first layer only ϵ approximating the basis of degree $p/2$ polynomials.
- We hope that by training the first hidden layer that the approximation error improves from $\epsilon \to 0$.

Bigger Questions

- Is $\mathrm{poly}(d)$ possible?

Bigger Questions

- Is $\mathrm{poly}(d)$ possible?
  A: There are reasons to believe that $d^{p/2}$ is a fundamental limit by analogy to tensor completion/sensing. SOS can get at best $d^{p/2}$ and conditional on hypergraphic planted clique (Luo and Zhang). I am not sure how much I believe this conjectured lower bound.

Bigger Questions

- What is a learnable function class for deeper teacher networks? I was thinking

$$f^*(x) = \sum_{s=1}^{r} \alpha_s (\beta_s^\top g_s^*(x))^{p_s},$$

  where $|\alpha_s| \leq 1$, $\mathbb{E}[(\beta_s^\top x)^{2p_s}] \leq 1$, $p_s \leq p$, and g_s^* is coordinatewise low rank polynomial.

- Easier to learn algorithmically (but I still don't know how to prove):

$$f^*(x) = \sum_{s} v_s^\top g_s^*(x) + \sum_{s=1}^{r} \alpha_s (\beta_s^\top g_s^*(x))^{p_s}$$