Learning One-hidden-layer Neural Networks With Landscape Design

Tengyu Ma

Facebook AI Research

Based on joint work with Rong Ge (Duke) and Jason D. Lee (USC)

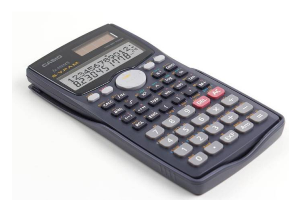
Interfaces Between Users and Optimizers?

Users

Optimization Researchers

function f

Solution



gradient descent
local search

Convex
relaxation
+ Rounding

Interfaces Between Users and Optimizers?

Users

function f

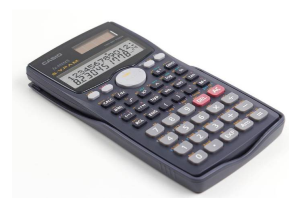
$$f = f_1 + \cdots + f_n$$

f_i is convex, smooth

condition number, $\cdots$

Solution

Optimization Researchers



Stochastic gradient descent

SAGA, SDCA, SVRG, …

Optimization in Machine Learning: New Interfaces?

Users

function f

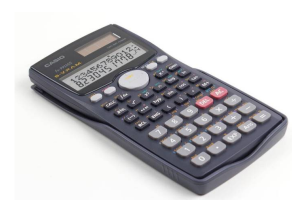
Optimization Researchers

Too hard, can you change the function?

Is this function easy for me?

Well, let me try a new model and a mew loss …

A new function f'



Solution for f'

(No rounding)

Stochastic gradient descent

NB: In learning,
model: $\hat{y} = g_\theta(x)$
loss: $f(\theta) = \mathbb{E}[\ell(y, g_\theta(x))]$

Optimization in Machine Learning: New Interfaces?

Users

Optimization Researchers

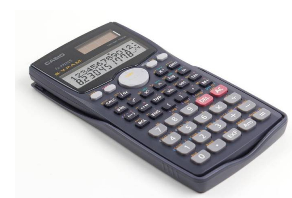
function f

Too hard, can you change the function?

Is this function easy for me?

Well, let me try a new model and a mew loss …

A new function f'



[ReLU, over-parameterization, batch normalization, residual networks ….]

Solution for f'

(No rounding)

Stochastic gradient descent

Possible Paradigm for Optimization Theory in ML?

- Identify a family $\mathcal{F}$ of tractable functions

$$\mathcal{F} = \{f: \text{all (or most) local minma are approximate global minima}\}$$

- Decide whether a function belongs to the family $\mathcal{F}$

Analysis techniques: linear algebra + probability, Kac-Rice formula, …

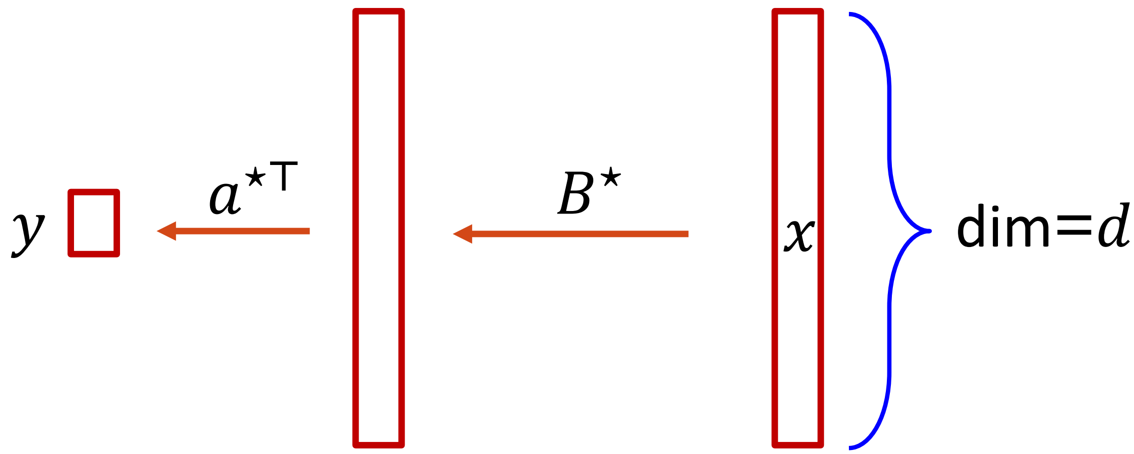
- Design new models and objective functions that are provably in $\mathcal{F}$

Some recent progress in simplified settings: [Hardt-M.-Recht'16, Soudry-Carmon'16, Liang-Xie-Song'17, Hardt-M.'17, Ge-Lee-M.'17]

NB: we also need to care about generalization error (but not in this talk)

This Talk: New Objective for Learning One-hidden-layer Neural Networks

- Assume data (x, y) satisfies
$$y = a^{\star\top}\sigma(B^{\star}x) + \xi$$
- Assume data x from Gaussian distribution
- Goal: learn a function that predicts y given x



- (σ = ReLU for all experiments in the talk)

$\text{Label } y = a^{\star\top}\sigma(B^{\star}x) + \xi$

The Straightforward Objective

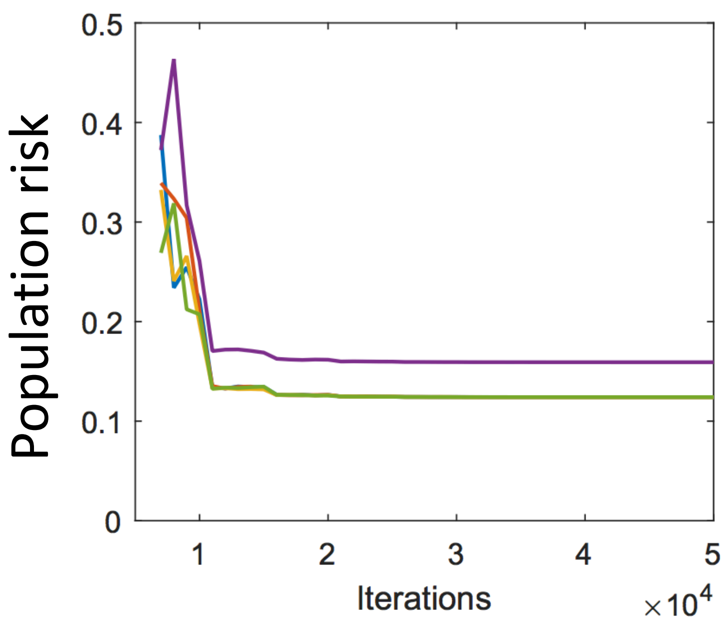
Our prediction

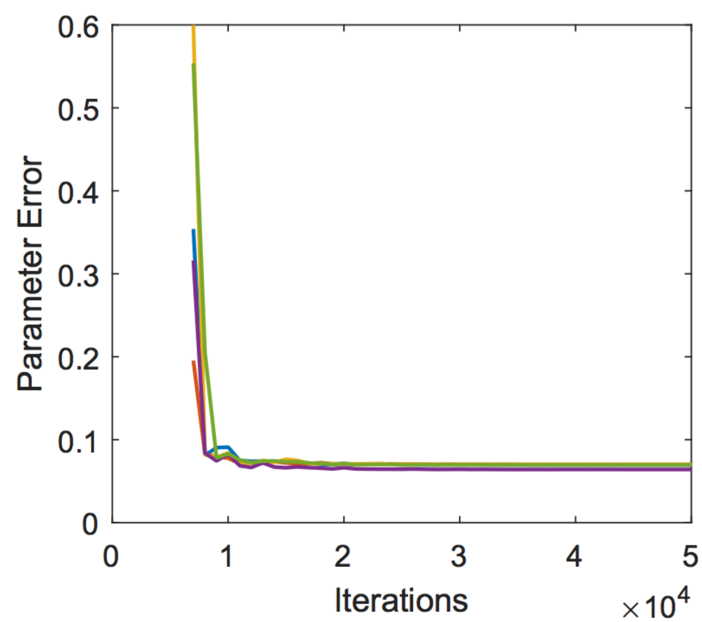
$$\hat{y} = a^{\top}\sigma(Bx)$$

- Loss function (population)

$$\mathbb{E}[(y - \hat{y})^2]$$

The Straightforward Objective Fails





- $d = 50$
- $a^\star = \mathbf{1}$ and assumed to be known
- $B^\star = I_{50\times 50}$
- $\xi = 0$
- fresh samples every iteration

dist$(B, B^\star)$ measured by a surrogate error $\geq \epsilon$

$\Leftrightarrow$ A row or a column of B is ϵ-far away from the natural basis in infinity norm

Related Work

- Non-overlapping filters (rows of $B^\star$ have disjoint supports) [Brutzkus-Globerson'17, Tian'17]
- Initialization is sufficiently close to $B^\star$ in spectral norm [Li-Yuan'17]
  - NB: the bad local min found is very far from $B^\star$ in spectral norm but close in infinity norm
- Kernel-based methods [Zhang et al.'16,'17]
- Tensor decomposition followed by local improvement algorithms [Janzamin et al.'15, Zhong et al.'17]
- Empirical solution: over-parameterization [Livni et al.'14]

Users

Well, let me try a new model and a new loss …

Main goal of this this talk

Optimization Researchers

Is this function easy for me?

Next slide: understand this better?

An Analytic Formula

Label $y = a^{\star\top}\sigma(B^\star x) + \xi$

Loss $f(a, B) = \mathbb{E}[||y - a^\top\sigma(Bx)||^2]$

Theorem 1: suppose the rows of B are unit vectors and $x \sim N(0, I)$

$$f(a, B) = \sum_{k\in\mathbb{N}} \hat{\sigma}_k^2 \left\| \sum_{i\in[m]} a_i^\star b_i^{\star\otimes k} - \sum_{i\in[m]} a_i b_i^{\otimes k} \right\|_F^2 + \text{const}\,.$$

- $\hat{\sigma}_k$ = the Hermite coefficient of σ
  - h_k = k-th normalized Hermite polynomial
  - $\hat{\sigma}_k := \mathbb{E}[\sigma(x)h_k(x)]$

$$B = \begin{bmatrix} b_1^\top \\ \vdots \\ b_m^\top \end{bmatrix} \quad B^\star = \begin{bmatrix} b_1^{\star\top} \\ \vdots \\ b_m^{\star\top} \end{bmatrix}$$

$$f(a,B) = \sum_{k\in\mathbb{N}} \hat{\sigma}_k^2 \underbrace{\left\| \sum_{i\in[m]} a_i^\star b_i^{\star \otimes k} - \sum_{i\in[m]} a_i b_i^{\otimes k} \right\|_F^2}_{:= f_k} + \text{const}\,.$$

- $f_0 = (\sum a_i^\star - \sum a_i)^2$
  - Convex, not identifiable
- $f_1 = ||\sum a_i^\star b_i^\star - \sum a_i b_i||^2$
  - No spurious local min, not identifiable
- $f_2 = ||\sum a_i^\star b_i^\star b_i^{\star\top} - \sum a_i b_i b_i^\top||_F^2$
  - No spurious local min? not identifiable
- $f_4 = ||\sum a_i^\star b_i^{\star\otimes 4} - \sum a_i b_i^{\otimes 4}||_F^2$
  - ∃ bad saddle point, identifiable

Each f_k solves a tensor decomposition problem

More difficult landscape? Stronger identifiability

A sweat spot?
A: yes, to some extent

Label $y = a^{\star\top} \sigma(B^\star x) + \xi$

New Loss Function

$$f_\gamma(a, B) = \mathbb{E}[||y - a^\top \gamma(Bx)||^2]$$

$$f_\gamma(a, B) = \sum_{k \in \mathbb{N}} \left\| \hat{\sigma}_k \sum_{i \in [m]} a_i^\star {b_i^\star}^{\otimes k} - \hat{\gamma}_k \sum_{i \in [m]} a_i b_i^{\otimes k} \right\|_F^2$$

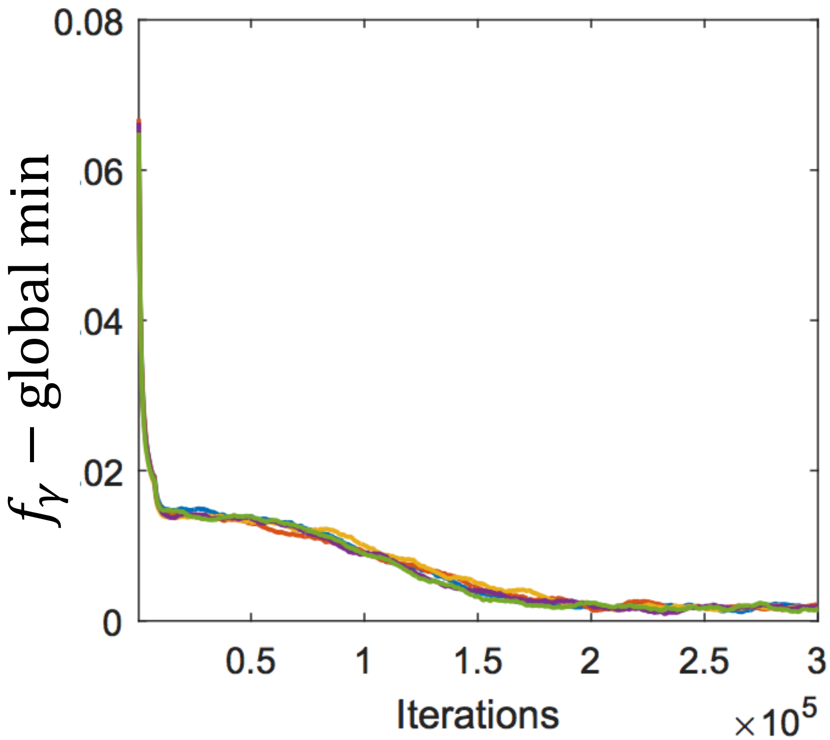
- Choosing γ such that $\hat{\gamma}_2 = \hat{\sigma}_2, \hat{\gamma}_4 = \hat{\sigma}_4$, and $\hat{\gamma}_k = 0$ for $k \neq 2,4$

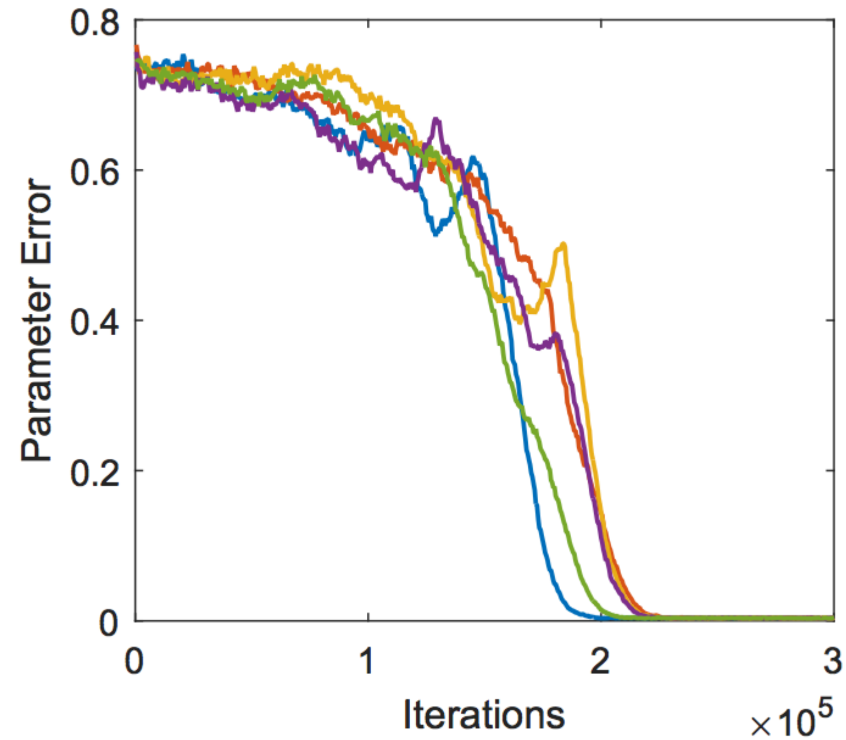
$$f_\gamma(a, B) = \hat{\sigma}_2^2 f_2 + \hat{\sigma}_4^2 f_4 + \text{const}$$

- Hope: the landscape of f_γ is better (and easier to analyze)
- 😃 Now empirically it works!
- 😔 Still we don't know how to analyze (more or provable alg. later)

$\text{Label } y = a^{\star\top}\sigma(B^\star x) + \xi$

$\text{Loss } f_\gamma(a, B) = \mathbb{E}[||y - a^\top \gamma(Bx)||^2]$





- $\sigma = \text{ReLU}$
- $d = 50$
- $a = \mathbf{1}$ and assumed to be known
- $B^\star = I_{50\times 50}$
- fresh samples every iteration

$\text{dist}(B, B^\star)$ measured by a surrogate error $\geq \epsilon$

$\Leftrightarrow$ A row or a column of B is ϵ-far away from the natural basis

Provable Non-convex Optimization Algorithms?

- Key lemma for proving Theorem 1

$$\mathbb{E}\left[y \cdot h_k(b_i^\top x)\right] = \hat{\sigma}_k \sum_{j \in [d]} a_j^\star \langle b_j^\star, b_i \rangle^k$$

- Extension (informal): for any polynomial p, there exists a function ϕ^p, such that

$$\mathbb{E}\left[y \cdot \phi^p(b_i, x)\right] = \sum_{j \in [d]} a_j^\star p(\langle b_j^\star, b_i \rangle)$$

- for any polynomial q over two variables, $\exists \phi^q$ s.t.

$$\mathbb{E}\left[y \cdot \phi^p(b_j, b_k, x)\right] = \sum_{j \in [d]} a_j^\star q(\langle b_j^\star, b_i \rangle, \langle b_j^\star, b_k \rangle)$$

- Next: find an objective that uses these gadgets, and have no spurious local minimum

An Objective Function with Guarantees

$$\min G(B) = \sum_{i\in[d]} a_i^\star \sum_{j\neq k} \langle b_i^\star, b_j\rangle^2 \langle b_i^\star, b_k\rangle^2 - \mu \sum_{i,j} a_i^\star \langle b_i^\star, b_j\rangle^4$$

$$s.t \quad \|b_i\|^2 = 1, \forall i$$

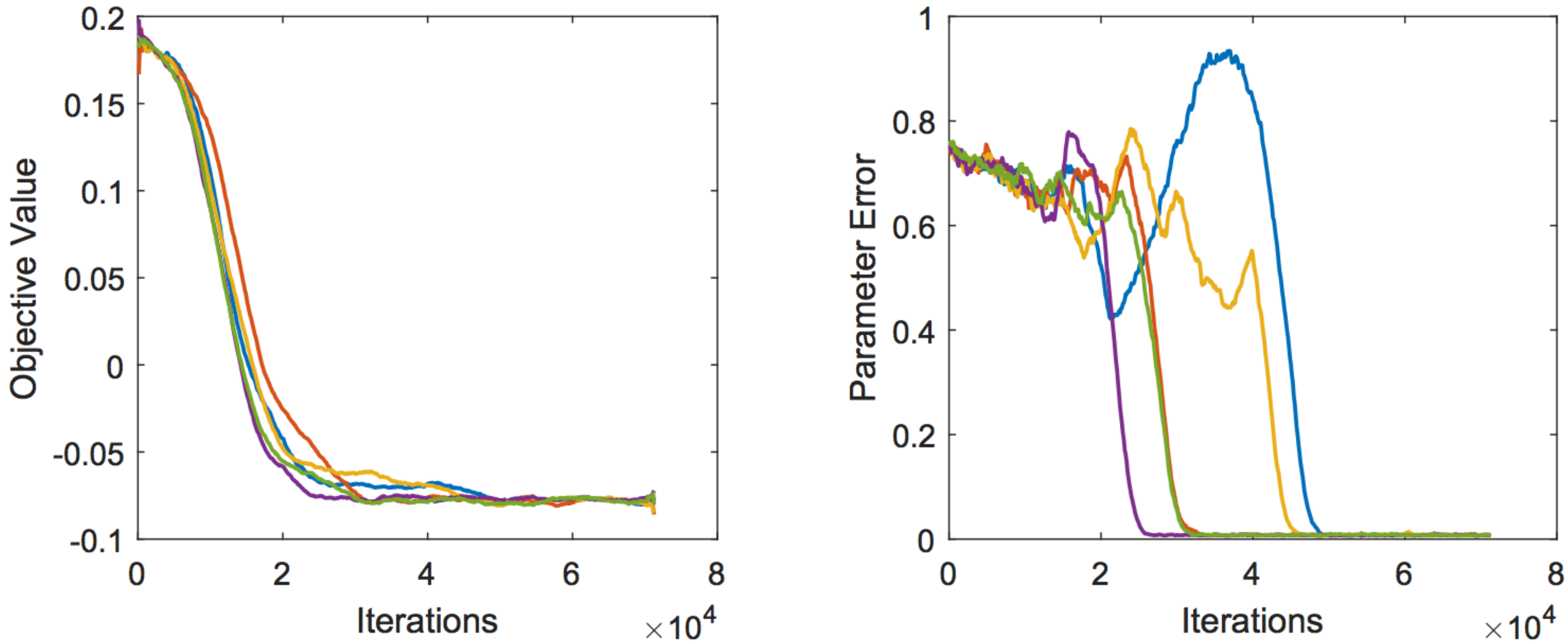
Theorem: assume $a^\star \geq 0$, $B^\star$ is orthogonal

1. $G(B)$ can be estimated via samples: $G(B) = \mathbb{E}[y \cdot \phi(B, x)]$

2. A global minimum of G is equal to $B^\star$ up to permutation and scaling of the rows

3. All the local minima of G are global minima

- Inspired by GHJY'15, which proved the case when $\mu = 0$ and $a_i^\star = 1$
- Can be extended to non-singular $B^\star$
- Limitation: $B^\star$: $\mathbb{R}^d \to \mathbb{R}^m$ with $m \leq d$



- Caveat: need huge batch size and training datasets

Conclusion

- Landscape design: designing new models and objectives with good landscape properties
- This paper: one first step for simplified neural nets

Open questions:

- Sample efficiency: killing higher-order term seems to lose information
  - Best empirical result: using $|\cdot|$ for training ReLU
- Beyond Gaussian inputs
- Understanding over-parameterization
- More techniques for analyzing optimization landscape

Thank you!