Fixing The NTK: From Neural Network Linearizations to Exact Convex Programs

Rajat Vadiraj Dwaraknath, Tolga Ergen, Mert Pilanci
Stanford University

Neurips 2023

Introduction

We combine three ideas in this work:

1. Convex programs for ReLU networks

2. Multiple Kernel Learning (MKL)

3. The Neural Tangent Kernel (NTK)

Supervised Learning Problem

- Data: inputs $X \in \mathbb{R}^{n \times d}$, scalar labels $y \in \mathbb{R}^n$
- Two layer ReLU network: $f(x) = \sum_{j=1}^{m} (x^T \mathbf{w}_j^{(1)})_+ w_j^{(2)}$
- Non-convex training with square loss and ℓ_2 regularization:

$$\min_{\mathbf{w}^{(1)}, w^{(2)}} \frac{1}{2} \|f(X) - y\|_2^2 + \lambda \sum_{j=1}^{m} (\|\mathbf{w}_j^{(1)}\|_2^2 + |w_j^{(2)}|^2)$$

Convex Programs for ReLU Networks

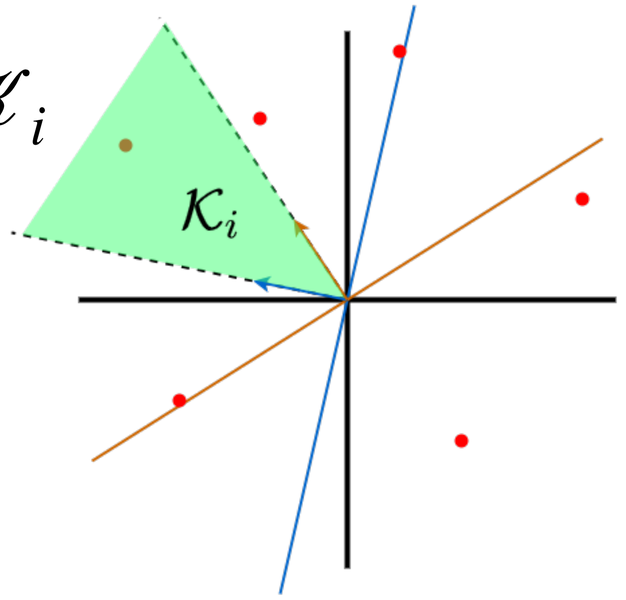
- Equivalent **Convex Program [Pilanci & Ergen 2020]:**

$$\min_{\mathbf{w},\mathbf{w}'} \left\| \sum_{i=1}^{p} D_i X(\mathbf{w}_i - \mathbf{w}'_i) - y \right\|_2^2 + \lambda \sum_{i=1}^{p} (\|\mathbf{w}_i\|_2 + \|\mathbf{w}'_i\|_2)$$

Group Lasso

$$\text{if } u \in \mathcal{K}_i \text{, then } (Xu)_+ = D_i X u$$

$$\textbf{subject to} \quad \mathbf{w}_i \in \mathcal{K}_i, \mathbf{w}'_i \in \mathcal{K}_i$$



- $D_i \in \{\textbf{diag}(\mathbf{1}(Xu \geq 0)) : u \in \mathbb{R}^d\}$
- $\mathcal{K}_i = \{\mathbf{w} \in \mathbb{R}^d : \textbf{diag}(\mathbf{1}(Xu \geq 0)) = D_i\}$

Gated ReLU Networks

- Relax cone constraints!

$$\min_{\mathbf{w}} \left\| \sum_{i=1}^{p} D_i X \mathbf{w}_i - y \right\|_2^2 + \lambda \sum_{i=1}^{p} \|\mathbf{w}_i\|_2$$

- Corresponds to Gated ReLU network **[MSP 2022]**:

$$f(x) = \sum_{j=1}^{m} \mathbf{1}(x^T \mathbf{g}_j \geq 0) \cdot x^T \mathbf{w}_j^{(1)} w_j^{(2)}$$

- Gates $\mathbf{g}_j$ are **fixed.**

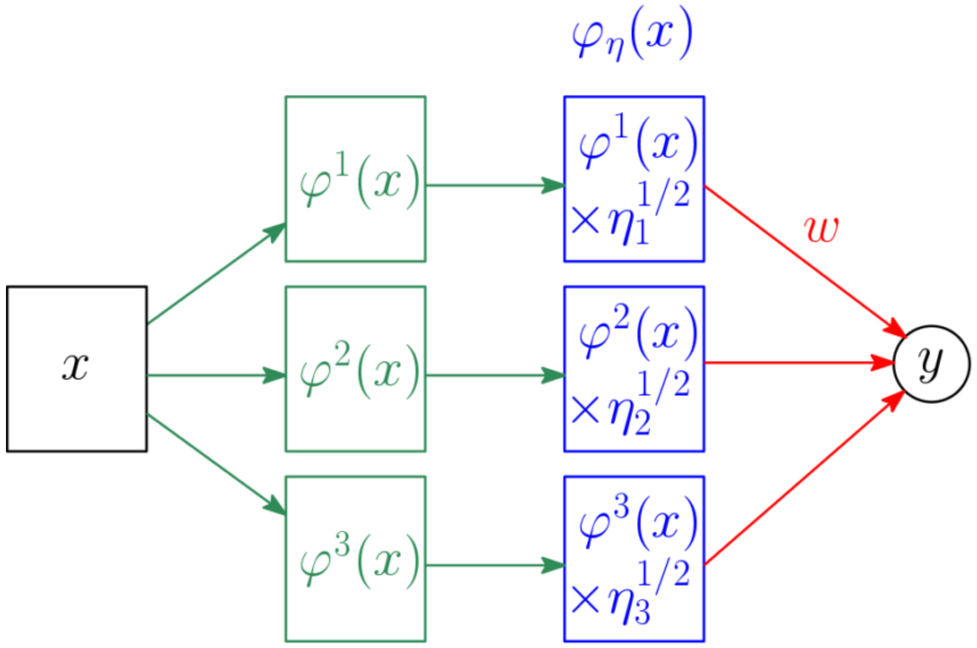
Multiple Kernel Learning

- Gated ReLU Convex program equivalent to MKL:

$$\min_{\eta \in \Delta_p, \mathbf{w}} \left\| \sum_{i=1}^{p} \sqrt{\eta_i} (D_i X) \mathbf{w}_i - y \right\|_2^2 + \lambda \sum_{i=1}^{p} \|\mathbf{w}_i\|_2^2$$

Ridge

- Each $D_i X$ is a feature matrix with weight $\sqrt{\eta_i}$
- Corresponding kernels are $K_i = D_i X X^T D_i$ with weight η_i
- **Gated ReLU does MKL with Masking Kernels!**



Neural Tangent Kernel

- Taylor expand the neural network

$$f(x, \theta) \approx f(x, \theta_0) + \nabla_\theta f(x, \theta_0)^T (\theta - \theta_0)$$

- Feature map: $\phi(x) = \nabla_\theta f(x, \theta_0)$

- **Neural Tangent Kernel:** $H(x, y) = \nabla_\theta f(x, \theta_0)^T \nabla_\theta f(y, \theta_0)$

- Deterministic in **infinite width limit** even if θ_0 is random

Putting It All Together

- $\mathbf{H} \in \mathbb{R}^{n \times n}$: infinite width NTK of Gated ReLU on training data

$\mathbf{H}$ lies in the MKL search space: $\mathbf{H} = \sum_{i=1}^{p} \tilde{\eta}_i K_i$ where

$\tilde{\eta}_i = \mathbb{P}[h \in \mathscr{K}_i]$ for $h \sim \mathcal{N}(0, \mathbf{I})$.

- MKL learns the optimal **data-dependent** kernel and fixes the NTK!