Revealing the Structure of Deep Neural Networks via Convex Duality

ICML 2021

Tolga Ergen & Mert Pilanci

July 19, 2021

Stanford University

Deep Learning Revolution

Deep Learning Revolution

Deep learning models:

- often provide the best performance due to their large capacity
  - challenging to train

Deep Learning Revolution

Deep learning models:

- often provide the best performance due to their large capacity
  - challenging to train
- are complex black-box systems based on non-convex optimization
  - hard to interpret what the model is actually learning

Prior Work on Regularized Deep Learning Training Problems

	Width (m)	**Assumption**	**Depth** (L)	**# of outputs** (K)
(Savarese et al., 2019)	*1*	1D data ($d = 1$)	2	7 ($K = 1$)
(Parhi and Nowak, 2019)	*1*	1D data ($d = 1$)	2	7 ($K = 1$)
(Ergen and Pilanci, 2020a,b)	nite	rank-one/whitened	2	3 (K 1)
Our results	nite	rank-one/whitened or BatchNorm	L 2	3 (K 1)

Prior Work on Regularized Deep Learning Training Problems

	Width (m)	**Assumption**	**Depth** (L)	**# of outputs** (K)
(Savarese et al., 2019)	*1*	1D data (d = 1)	2	7 (K = 1)
(Parhi and Nowak, 2019)	*1*	1D data (d = 1)	2	7 (K = 1)
(Ergen and Pilanci, 2020a,b)	nite	rank-one/whitened	2	3 (K 1)
Our results	nite	rank-one/whitened or BatchNorm	L 2	3 (K 1)

Optimal solution for L-layer ReLU networks is given by piecewise linear splines for any L 2.

Figure 1: One dimensional interpolation using L-layer ReLU networks

Warmup: Two-layer Linear Networks

$\mathbf{X} \in \mathbb{R}^{n \times d}$: Data matrix; $\mathbf{y} \in \mathbb{R}^{n}$: Label vector

$\mathbf{W}_l \in \mathbb{R}^{m_{l-1} \times m_l}$: l^{th} layer weight matrix

$L(\cdot;\cdot)$: Arbitrary convex loss function

$\lambda > 0$: Regularization coefficient

$f_{\theta,L}(\mathbf{X})$: Output of an L-layer network

- **Model:** $f_{\theta,2}(\mathbf{X}) = \mathbf{X}\mathbf{W}_1\mathbf{w}_2$

Warmup: Two-layer Linear Networks

$\mathbf{X} \in \mathbb{R}^{n \times d}$: Data matrix; $\mathbf{y} \in \mathbb{R}^{n}$: Label vector

$\mathbf{W}_l \in \mathbb{R}^{m_{l-1} \times m_l}$: l^{th} layer weight matrix

$L(\cdot, \cdot)$: Arbitrary convex loss function

$\beta > 0$: Regularization coefficient

$f_{\theta,L}(\mathbf{X})$: Output of an L-layer network

- **Model:** $f_{\theta,2}(\mathbf{X}) = \mathbf{X}\mathbf{W}_1\mathbf{w}_2$
- **Optimization problem:**

$$\min_{\mathbf{W}_1, \mathbf{w}_2} L(f_{\theta,2}(\mathbf{X}), \mathbf{y}) + \beta \left(\|\mathbf{W}_1\|_F^2 + \|\mathbf{w}_2\|_2^2\right)$$

Warmup: Two-layer Linear Networks

$\mathbf{X} \in \mathbb{R}^{n \times d}$: Data matrix, $\mathbf{y} \in \mathbb{R}^{n}$: Label vector

$\mathbf{W}_l \in \mathbb{R}^{m_{l-1} \times m_l}$: l^{th} layer weight matrix

$\mathcal{L}(\cdot,\cdot)$: Arbitrary convex loss function

$\beta > 0$: Regularization coefficient

$f_{\theta,L}(\mathbf{X})$: Output of an L-layer network

- **Model:** $f_{\theta,2}(\mathbf{X}) = \mathbf{X}\mathbf{W}_1\mathbf{w}_2$
- **Optimization problem:**

$$\min_{\mathbf{W}_1,\mathbf{w}_2} \mathcal{L}(f_{\theta,2}(\mathbf{X}),\mathbf{y}) + \beta\left(\|\mathbf{W}_1\|_F^2 + \|\mathbf{w}_2\|_2^2\right)$$

- **Optimal hidden layer weight:** $\mathbf{w}_1^* = \frac{\mathbf{X}^T P_{\mathbf{X},\beta}(\mathbf{y})}{\|\mathbf{X}^T P_{\mathbf{X},\beta}(\mathbf{y})\|_2}$
where $P_{\mathbf{X},\beta}(\cdot)$ projects to $\{\mathbf{u} \in \mathbb{R}^n \mid \|\mathbf{X}^T\mathbf{u}\|_2 \le \beta\}$.

Deep Linear Networks

- **Model:** $f_{\,;L}(\mathbf{X}) = \sum_{j=1}^{m} \mathbf{X}\mathbf{W}_{1;j}\mathbf{W}_{2;j} \cdots \mathbf{w}_{L;j}$

Deep Linear Networks

- **Model:** $f_{;L}(\mathbf{X}) = \sum_{j=1}^{m} \mathbf{X}\mathbf{W}_{1;j}\mathbf{W}_{2;j} \dots \mathbf{w}_{L;j}$
- **Optimization problem:**

$$\min_{\mathbf{W}_1;\mathbf{w}_2} L(f_{;L}(\mathbf{X}); \mathbf{y}) + \sum_{j=1}^{m} \sum_{l=1}^{L} \|\mathbf{W}_{l;j}\|_F^2$$

- **Model:** $f_{,L}(\mathbf{X}) = \sum_{j=1}^{m} \mathbf{X}\mathbf{W}_{1,j}\mathbf{W}_{2,j}\ldots\mathbf{w}_{L,j}$
- **Optimization problem:**

$$\min_{\mathbf{W}_1,\mathbf{w}_2} L(f_{,L}(\mathbf{X}),\mathbf{y}) + \sum_{j=1}^{m}\sum_{l=1}^{L} \|\mathbf{W}_{l,j}\|_F^2$$

- **Optimal hidden layer weights:**

$$\mathbf{W}_{l,j} = \begin{cases} t_j \frac{\mathbf{X}^T P_{\mathbf{X},}(\mathbf{y})}{\|\mathbf{X}^T P_{\mathbf{X},}(\mathbf{y})\|_2} {}_{1,j}^{T} & \text{if } l = 1 \\ t_j \, {}_{l-1,j} {}_{l,j}^{T} & \text{if } 1 < l \le L-2 \\ {}_{L-2,j} & \text{if } l = L-1 \end{cases},$$

where $\| \,_{l,j}\|_2 = 1$, $P_{\mathbf{X},}(\cdot)$ projects to $\left\{ \mathbf{u} \in \mathbb{R}^n \mid \|\mathbf{X}^T\mathbf{u}\|_2 \le t_j^{2-L} \right\}$ and $t_j = \|\mathbf{W}_{l,j}\|_F$.

Deep ReLU Networks

- **Model:** $f_{\ ;L}(\mathbf{X}) = \mathbf{A}_{L\ \ 1}\mathbf{w}_L$, where $\mathbf{A}_{l;j} = (\mathbf{A}_{l\ \ 1;j}\mathbf{W}_{l;j})_+;$, $\mathbf{A}_{0;j} = \mathbf{X}$, $8l;j$, and $(x)_+ = \max f0;xg$

- **Model:** $f_{\ ,L}(\mathbf{X}) = \mathbf{A}_{L-1}\mathbf{w}_L$, where $\mathbf{A}_{l,j} = (\mathbf{A}_{l-1,j}\mathbf{W}_{l,j})_+$, $\mathbf{A}_{0,j} = \mathbf{X}$, $\forall l, j$, and $(x)_+ = \max\{0, x\}$

Theorem

Let $\mathbf{X}$ be a rank-one matrix such that $\mathbf{X} = \mathbf{c}\mathbf{a}_0^T$, where $\mathbf{c} \in \mathbb{R}^n_+$ and $\mathbf{a}_0 \in \mathbb{R}^d$, then strong duality holds and the optimal weights are

$$\mathbf{W}_{l,j} = \frac{\ _{l-1,j}}{\|\ _{l-1,j}\|_2}\ _{l,j}^T,\ \forall l \in [L-2],\ \mathbf{w}_{L-1,j} = \frac{\ _{L-2,j}}{\|\ _{L-2,j}\|_2},$$

where $\ _{0,j} = \mathbf{a}_0$ and $\{\ _{l,j}\}_{l=1}^{L-2}$ is a set of vectors such that $\ _{l,j} \in \mathbb{R}^{m_l}_+$ and $\|\ _{l,j}\|_2 = t_j,\ \forall l \in [L-2], \forall j \in [m]$.

Deep ReLU Networks

- **Model:** $f_{;L}(\mathbf{X}) = \mathbf{A}_{L\ \ 1}\mathbf{w}_L$, where $\mathbf{A}_{l;j} = (\mathbf{A}_{l\ \ 1;j}\mathbf{W}_{l;j})_+;$, $\mathbf{A}_{0;j} = \mathbf{X}$, $\forall l;j$, and $(x)_+ = \max\{0;x\}$

Theorem

Let $\mathbf{X}$ be a rank-one matrix such that $\mathbf{X} = \mathbf{c}\mathbf{a}_0^T$, where $\mathbf{c} \in \mathbb{R}_+^n$ and $\mathbf{a}_0 \in \mathbb{R}^d$, then strong duality holds and the optimal weights are

$$\mathbf{W}_{l;j} = \frac{_{l\ \ 1;j}}{\|_{l\ \ 1;j}\|_2}\ {}^T_{l;j};\ \forall l \in [L\ \ \ 2];\ \mathbf{w}_{L\ \ 1;j} = \frac{_{L\ \ 2;j}}{\|_{L\ \ 2;j}\|_2};$$

where $_{0;j} = \mathbf{a}_0$ and $\{_{l;j}\}_{l=1}^{L\ \ 2}$ is a set of vectors such that $_{l;j} \in \mathbb{R}_+^{m_l}$ and $\|_{l;j}\|_2 = t_j^{\ };\ \forall l \in [L\ \ \ 2]; \forall j \in [m]$.

Corollary

For 1D data, i.e., $\mathbf{x} \in \mathbb{R}^n$, the optimal network output has kinks only at the input data points, i.e., the output function is in the following form: $f_{;L}(\hat{x}) = \sum_i\ (\hat{x}\ \ \ x_i)_+$. Therefore, the optimal network output is a linear spline interpolation.

Vector-output ReLU Networks

Theorem

Let $\{\mathbf{X}, \mathbf{Y}\}$ be a dataset such that $\mathbf{X}\mathbf{X}^T = \mathbf{I}_n$ and $\mathbf{Y} \in \mathbb{R}^{n \times K}$ is one-hot encoded, then a set of optimal solutions for the following regularized training problem

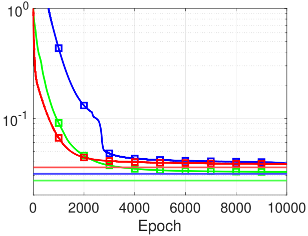
$$\min_{\theta \in \Theta} \frac{1}{2}\|f_{\theta,L}(\mathbf{X}) - \mathbf{Y}\|_F^2 + \frac{\beta}{2}\sum_{j=1}^{m}\sum_{l=1}^{L}\|\mathbf{W}_{l,j}\|_F^2$$

can be formulated as follows

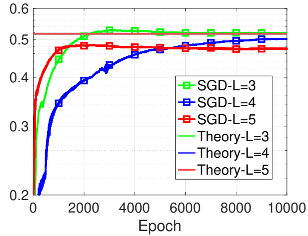
$$\mathbf{W}_{l,j} = \begin{cases} \frac{\phi_{l-1,j}}{\|\phi_{l-1,j}\|_2}\psi_{l,j}^T, & \text{if } l \in [L-1] \\ (\|\psi_{0,j}\|_2 - \beta)_+ \psi_{l-1,j}\mathbf{e}_r^T & \text{if } l = L \end{cases},$$

where $\psi_{0,j} = \mathbf{X}^T\mathbf{y}_j$, $\{\psi_{l,j}\}_{l=1}^{L-2}$ are vectors such that $\psi_{l,j} \in \mathbb{R}_+^{m_l}$, $\|\psi_{l,j}\|_2 = t_j$, and $\psi_{l,i}^T\psi_{l,j} = 0, \forall i \neq j$, Moreover, $\psi_{L-1,j} = \mathbf{e}_j$ is the j^{th} ordinary basis vector.

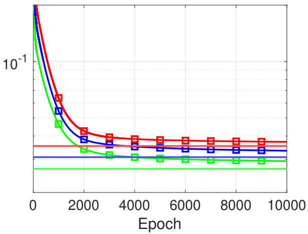
Numerical Results



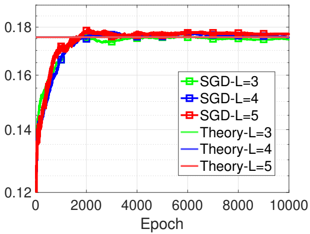
(a) MNIST-Training obj.



(b) MNIST-Test accuracy



(c) CIFAR10-Training obj.



(d) CIFAR10-Test accuracy

Figure 2: Training and test performance on whitened and sampled datasets.

Takeaways and Open Problems

- Optimal solutions to regularized deep neural network training problems can be explicitly characterized via convex analytic frameworks

Takeaways and Open Problems

- Optimal solutions to regularized deep neural network training problems can be explicitly characterized via convex analytic frameworks
- When the input data is whitened or rank-one, optimal layer weights of an L-layer deep ReLU network can be found the closed-form

Takeaways and Open Problems

- Optimal solutions to regularized deep neural network training problems can be explicitly characterized via convex analytic frameworks
- When the input data is whitened or rank-one, optimal layer weights of an L-layer deep ReLU network can be found the closed-form
- For 1D datasets, kinks of ReLU occur exactly at the input data so that the optimal network outputs linear spline interpolations

Takeaways and Open Problems

- Optimal solutions to regularized deep neural network training problems can be explicitly characterized via convex analytic frameworks
- When the input data is whitened or rank-one, optimal layer weights of an L-layer deep ReLU network can be found the closed-form
- For 1D datasets, kinks of ReLU occur exactly at the input data so that the optimal network outputs linear spline interpolations
- **Open problems:**
  - extension of the analysis to standard deep networks

Takeaways and Open Problems

- Optimal solutions to regularized deep neural network training problems can be explicitly characterized via convex analytic frameworks
- When the input data is whitened or rank-one, optimal layer weights of an L-layer deep ReLU network can be found the closed-form
- For 1D datasets, kinks of ReLU occur exactly at the input data so that the optimal network outputs linear spline interpolations
- **Open problems:**
  - extension of the analysis to standard deep networks
  - generalization properties of the optimal solutions

References

Ergen, T. and Pilanci, M. (2020a). Convex geometry and duality of over-parameterized neural networks. *arXiv preprint arXiv:2002.11219*.

Ergen, T. and Pilanci, M. (2020b). Convex geometry of two-layer relu networks: Implicit autoencoding and interpretable models. In Chiappa, S. and Calandra, R., editors, *Proceedings of the Twenty Third International Conference on Arti cial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 4024{4033, Online. PMLR.

Parhi, R. and Nowak, R. D. (2019). Minimum \norm" neural networks are splines. *arXiv preprint arXiv:1910.02333*.

Savarese, P., Evron, I., Soudry, D., and Srebro, N. (2019). How do infinite width bounded norm networks look in function space? *CoRR*, abs/1902.05040.