Peeking into the Black Box: Layerwise-Convex Training for Convolutional Neural Networks

Trenton Chang,[1] Raymond Lee[2]

[1]Department of Computer Science, Stanford University
[2]Department of Management Science and Engineering, Stanford University

Stanford
EE364B Final Project

Introduction

- **Deep neural networks** perform well on object detection, image classification, and other computer vision tasks
- **Deep neural network optimization** is extremely non-convex: 100s of composed non-linearities -> non-trivial to convexify
- **Non-convexity ->** No guarantees about optimization + low interpretability

Approach: Combine convex formulation of a small piece of a deep neural network + layer-wise training method.

Two-Layer CNNs: A Convex Formulation

Generic objective (Eq. 1): Minimize Mean Squared Error (MSE) w/ L2^2 regularization.

$$\text{minimize} \quad \frac{1}{2}\|\underbrace{f(x)}_{\text{Model output}} - \underbrace{y}_{\text{Expected or true output}}\|_2^2 + \frac{\lambda}{2}\sum_{i=1}^{m}(\underbrace{\|\mathbf{u}_i\|_2^2}_{\text{L2 reg; conv. layer}} + \underbrace{w_i^2}_{\text{L2 reg; linear layer}})$$

where $f(x) = \underbrace{\sum_{j=1}^{m} w_j}_{\text{Linear transform}} \underbrace{\sum_{k=1}^{K}(X_k\mathbf{u}_j)_+}_{\text{KxKConv2D: filters * image patches}}$ (non-convex).

Convex formulation (Eq. 2): Minimize MSE with Group-Lasso regularization. [Ergen and Pilanci 2021]

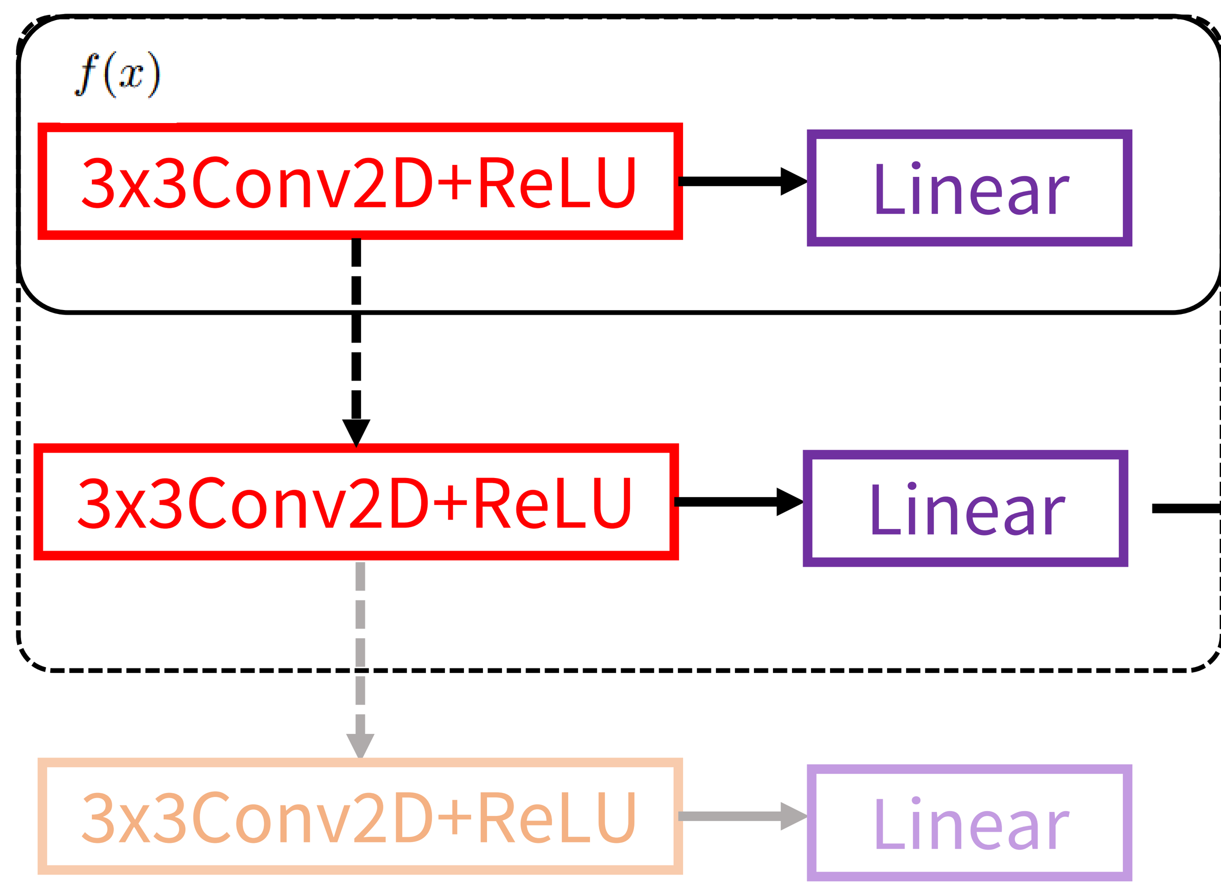
$$\text{minimize} \quad \frac{1}{2}\left\|\overbrace{\sum_{i=1}^{P_{conv}}\sum_{k=1}^{K}\mathbf{D}(S_i^k)X_k\mathbf{v}_i}^{\text{Model output}} - y\right\|_2^2 + \overbrace{\frac{\lambda}{2}\sum_{i=1}^{m}(\|\mathbf{v}_i\|_2)}^{\text{Group-lasso reg.}}$$

$$\text{subject to} \quad (\mathbf{I}_n - 2\mathbf{D}(S_i^k))X_k\mathbf{v}_i \le 0, \forall i, k$$

$\mathbf{D}(S_i^k)$: diagonal sign matrix of +/-1s

Main Experiment: Layer-Wise Training + Convex Form.

TL;DR: Stack Eq. 1/2 J times, J = {1, 2, 3, 4, 5}. [Belilovsky et. al. 2019]



Step 1: Train (via Eq. 1 or 2) a Conv+Linear model.

Step 2: Use Conv output as **input** to next layer, and Linear output as **prediction**.

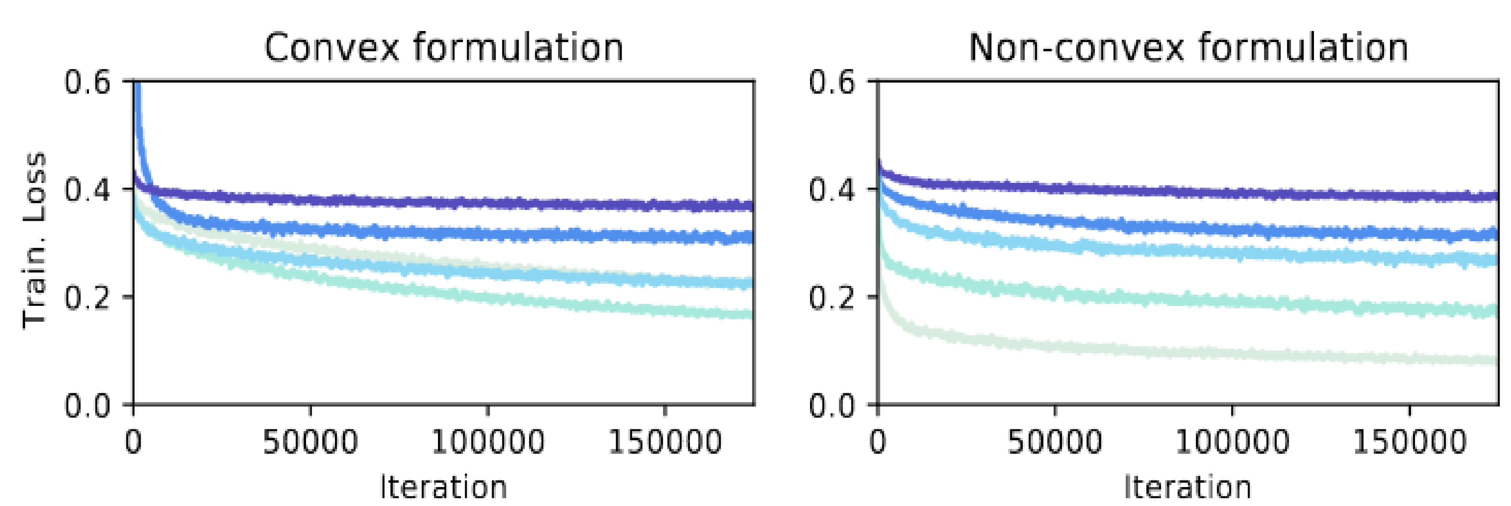
Step 3: Repeat Step 2 until J layers are trained.

*For simplicity, pooling operations are omitted here.

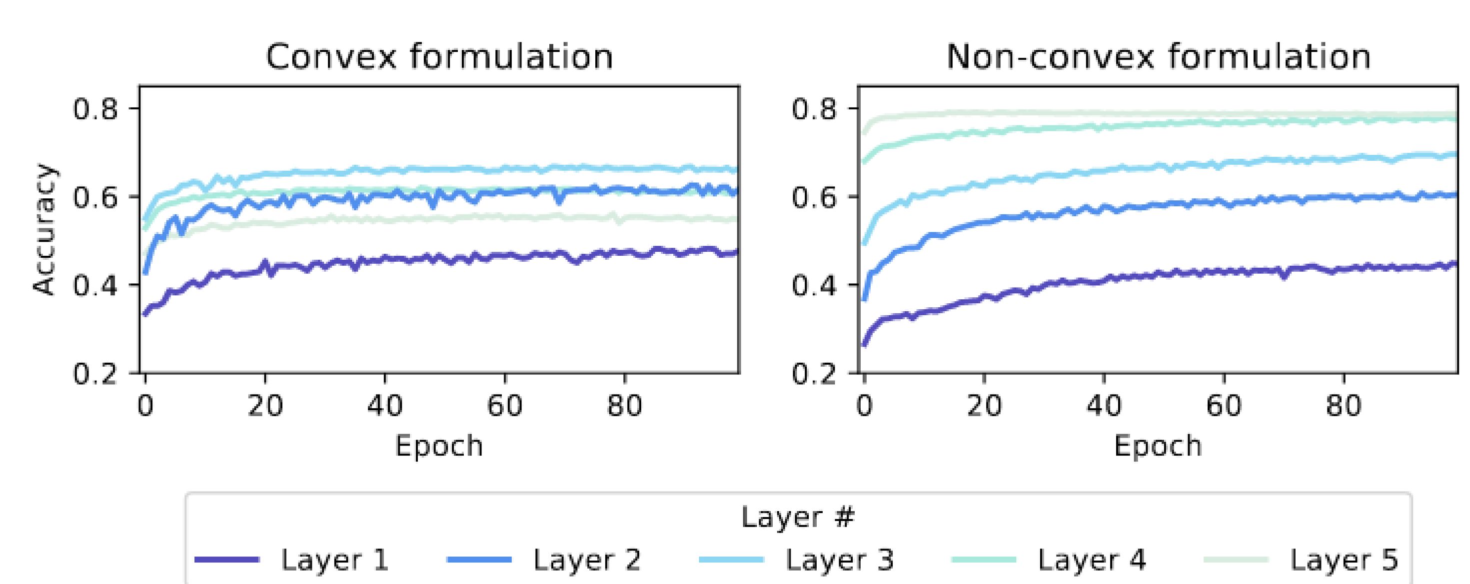
Layer-Wise Training: Works at Low Depths, Overfits at High Depths

Objective value: Each new **convex layer** (except last) brings down training loss.

Accuracy: Convex formulation outperforms non-convex form. for layers 1, 2; tied at layer 3.



Convex formulation fits training data better: Training loss at each layer (except last) is higher for non-convex formulation.

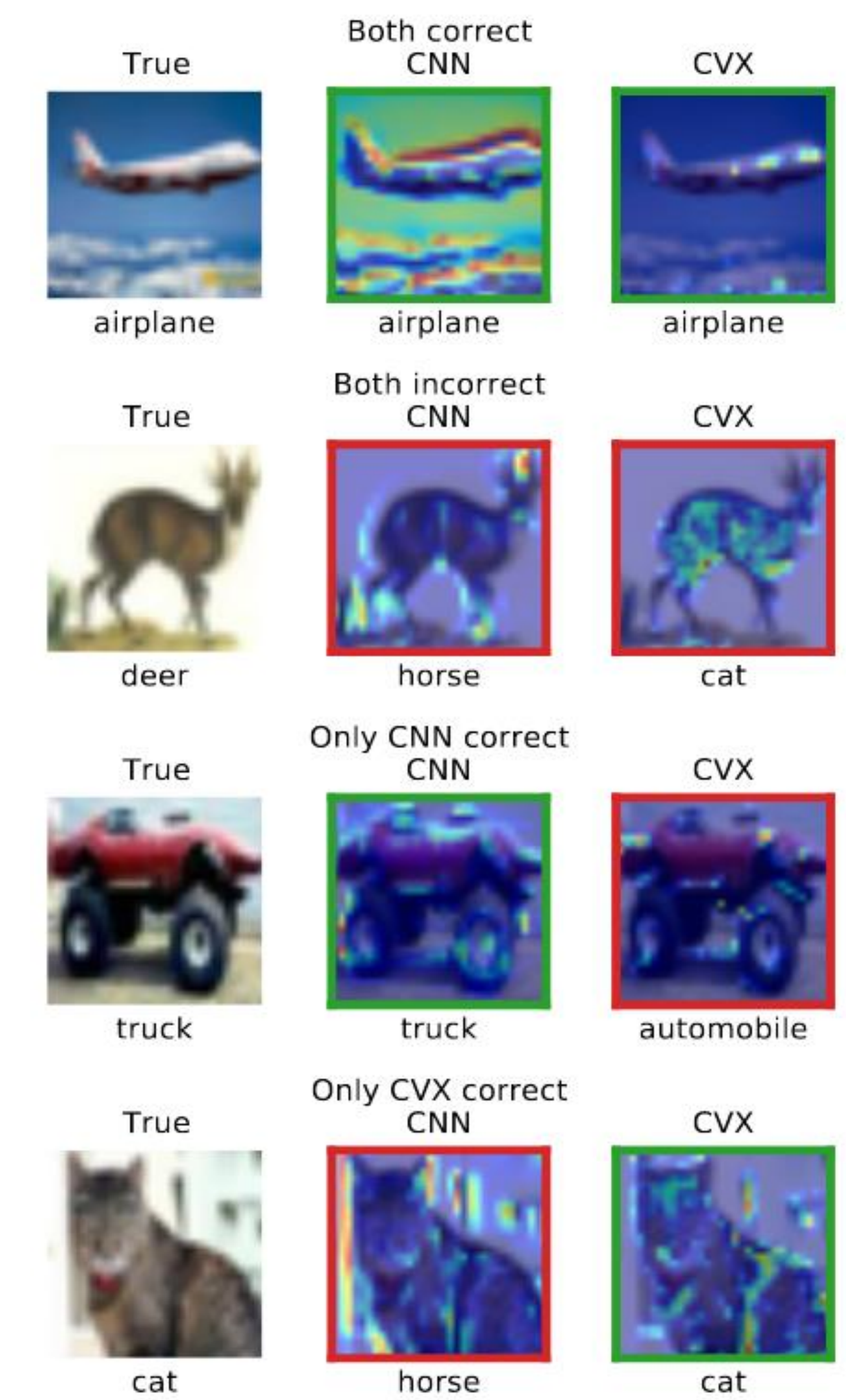


Convex formulation runs into overfitting ("good" on train, "bad" on test): test-accuracy plateaus at layer 4 and up even as training loss decreases.

*Lighter color = more advanced layer.

Qualitative Analysis of Solutions

Use GradCAM Visualization to show how much parts of image affect the prediction. [Selvaraju et. al. 2019]



(a) Grad-CAM, single-layer model.

1. **How intensely does the heatmap "light up?" The convex formulation** (CVX) has sparser solutions than non-convex (CNN) formulation **(right column; 1st row, 2nd from bottom, bottom).**

2. ***Where* in the image does the heatmap "light up?" Both formulations result in spurious correlations**: attention to clouds **(1st row),** attention to image background **(bottom row)**

Conclusion & Future Work

There is promise for scaling up convex formulations of CNN components. However, at deeper layers, **adding further convex layers results in overfitting, fitting the training data well but failing to generalize,** necessitating further work to condition layerwise-convex training at deeper layers. Possible next direction: testing learning rate scheduling (decay and warmup).

References

1. [Belilovsky et al., 2019] Belilovsky, E., Eickenberg, M., and Oyallon,E. (2019).Greedy Layerwise Learning Can Scale to ImageNet.arXiv:1812.11446 [cs, stat].
2. [Ergen and Pilanci, 2021a] Ergen, T. and Pilanci, M. (2021a).Implicit Convex Regularizers of CNN Architectures: Convex Optimization of Two-and Three-Layer Networks 3. in Polynomial Time.arXiv:2006.14798 [cs,stat]
3. [Selvaraju et al., 2019] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam,R., Parikh, D., and Batra, D. (2019). Grad-cam: Visual explanations from deep networks via gradient-based localization. International Journal ofComputer Vision, 128(2):336–359.

See full paper for comprehensive reference list. Special thanks to Tolga Ergen, Department of Electrical Engineering, Stanford University, for mentoring this project.