UVA CS 6316: Machine Learning : 2019 Fall
Course Project: Deep2Reproduce @
https://github.com/qiyanjun/deep2reproduce/tree/master/2019Fall

Visualizing the Loss Landscape of Neural Nets

Reproduced by: Haochuan Zhang, Yu Du

2019/12

Motivation

- Understand the effect of training parameters and network architectures on loss landscapes and the shape of minimizers
- Find the effect of loss landscapes on generalization
- Does loss landscape show significant non-convexity?

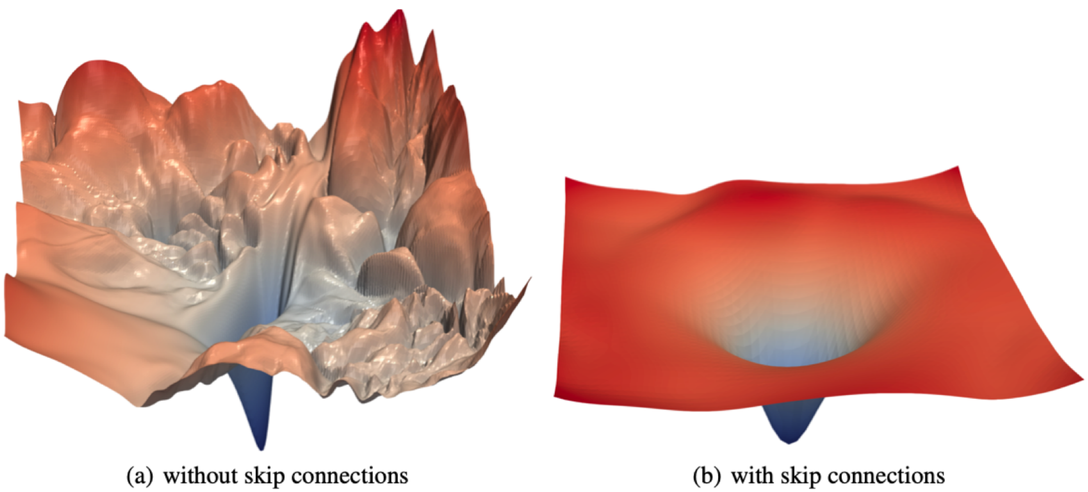


Figure 1: The loss surfaces of ResNet-56 with/without skip connections. The proposed filter normalization scheme is used to enable comparisons of sharpness/flatness between the two figures.

Background

- Trainability of neural nets is highly dependent on:
  - Network architecture
  - Optimizer
  - Variable initialization and etc.
- Globally optimal or near-optimal solutions can be found by common optimization methods for restricted network classes[2, 3, 4]
- Relationship between sharpness/flatness of local minima and generalization ability:
  - Small-batch SGD produces flat minimals that generalize well
  - Large-batch SGD produces sharp minimals and has poor generalization

Related Work

- 1-Dimensional Linear Interpolation by Goodfellow et al. [5]

  $$\theta(\alpha) = (1-\alpha)\theta + \alpha\theta'$$

  $$f(\alpha) = L(\theta(\alpha))$$

- Contour Plots & Random Directions

  $$f(\alpha, \beta) = L(\theta^* + \alpha\delta + \beta\eta)$$

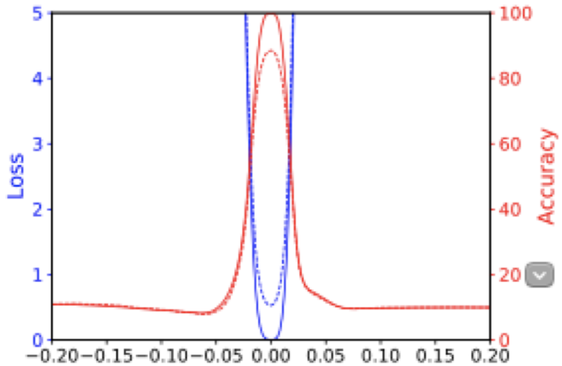
  - Explore the trajectories of minimization methods

Claim / Target Task

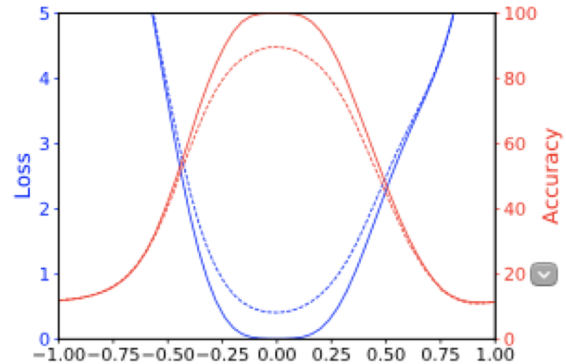
- 1D Linear Interpolation
  - hard to visualize non-convexities
  - does not consider batch normalization
- Contour Plots & Random Directions:
  - 2D case but computational burden is large causes low-resolution
  - Fails to capture the intrinsic geometry of loss surfaces
- Scale invariance in (rectified) network weights
  - Prevent meaningful comparisons between plots of different networks
- Sharp minimizers or flat minimizers generalize better?
  - The difference between sharp and flat minimizers
  - How to visualize?

An Intuitive Figure Showing WHY Claim

VGG9

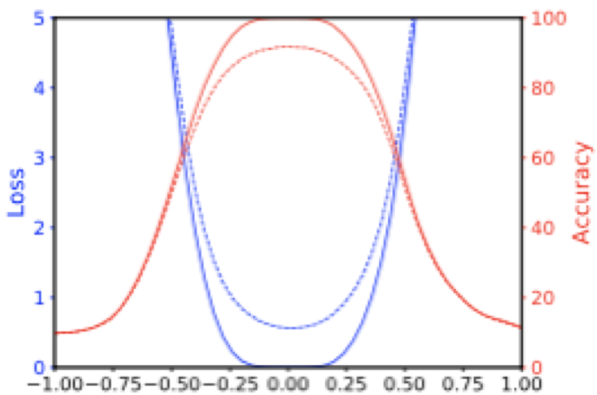


(b) SGD, 8192, 11.07%



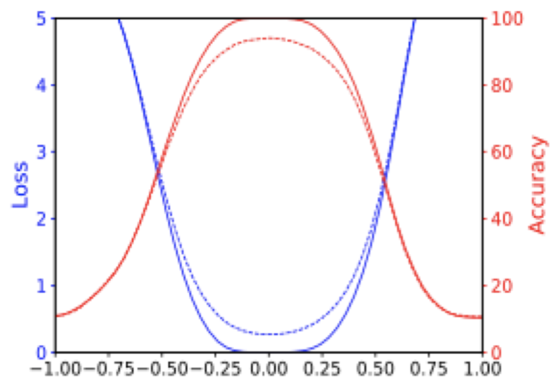
(f) SGD, 8192, 10.19%

No Normalization



(a) SGD, 128, 8.26%

Normalized



(e) SGD, 128, 5.89%

Proposed Solution

- Filter-Wise Normalization
  - Produce a random Gaussian direction vector *d*
  - *d* is dimensional compatible with θ.
  - Normalize each filter in *d* to have the same norm of corresponding filter in θ.
  - Will be applied to convolutional layers and fully connected layers
  - ps. *j* means *j*th filter in *i*th layer of *d*

$$d_{i,j} \leftarrow \frac{d_{i,j}}{\|d_{i,j}\|}\|\theta_{i,j}\|$$

- Explore the relationship between generalization and flatness/sharpness
- Explore different architecture effect

Implementation

- Prepare pretrained models with different parameters will be used
- Load models and extract parameters
- Setup the direction file and the image file in .h5 file
  - Filter normalization:

```
for d, w in zip(direction, weights):
    d.mul_(w.norm()/(d.norm() + 1e-10))
```

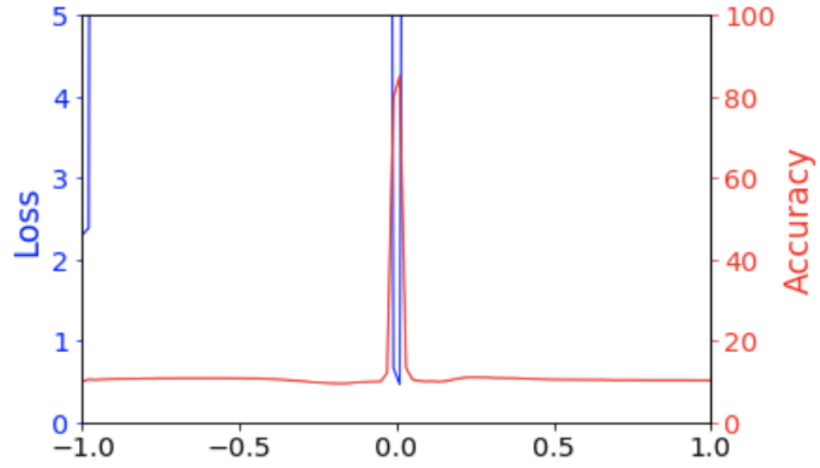
- Calculate loss values and accuracies: cross entropy
- Plot figures

Data Summary

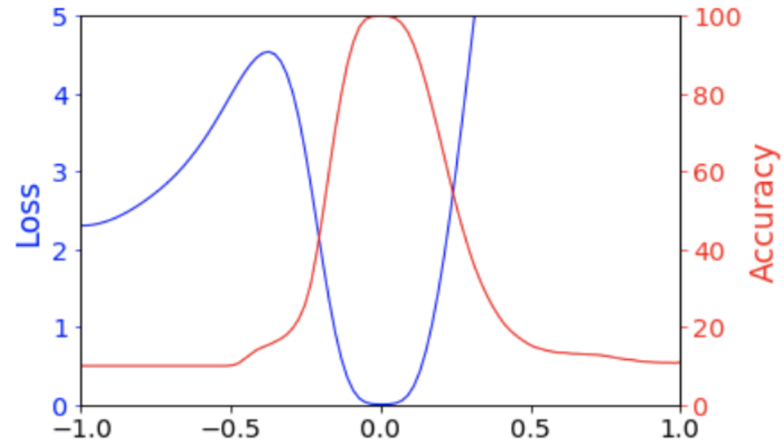
- Dataset
  - Cifar 10
- Pretrained Models
  - VGG-9
  - ResNet 56
  - ResNet 56 (no shortcut)

Batch size	128, 8192
Weight Decay	0, 0.0005
# of epoches	300
Learning Rate	0.1

Experimental Results & Analysis



VGG9, batch size=8192, weight decay=0.0005, no normalization test error=11.34%

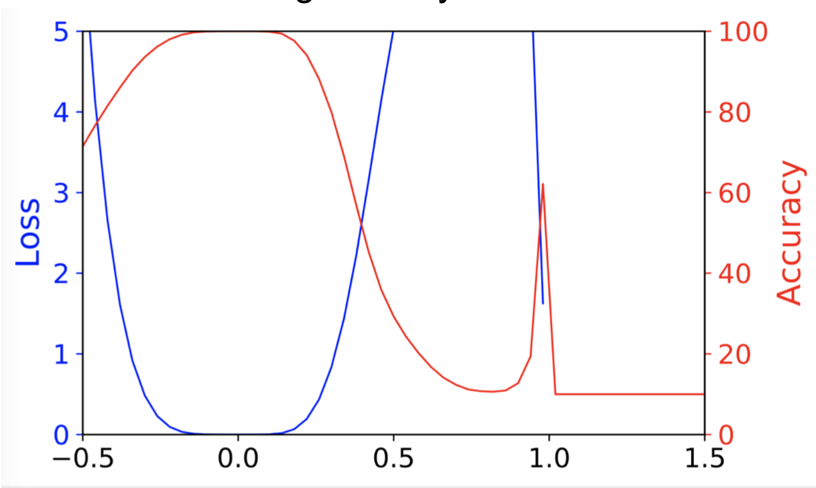


VGG9, batch size=8192, weight decay=0.0005, filter normalization test error = 10.47%

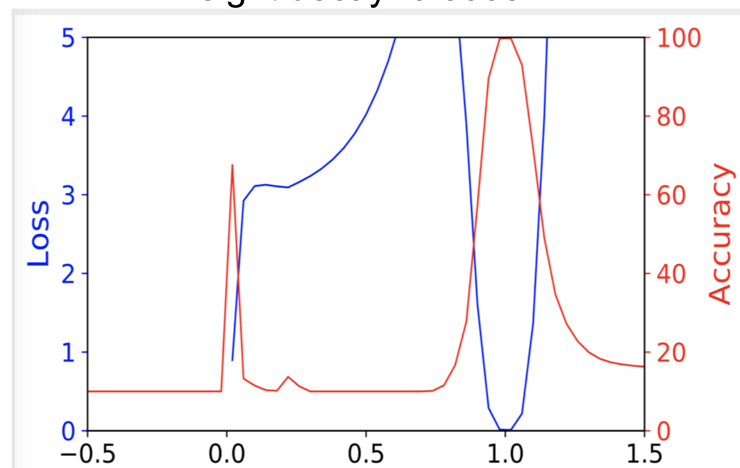
Filter-wise Normalization is more accurate.

Experimental Results & Analysis

Weight decay=0



Weight decay=0.0005



Test Error	7.53%	11.24%	6.21%	10.08%

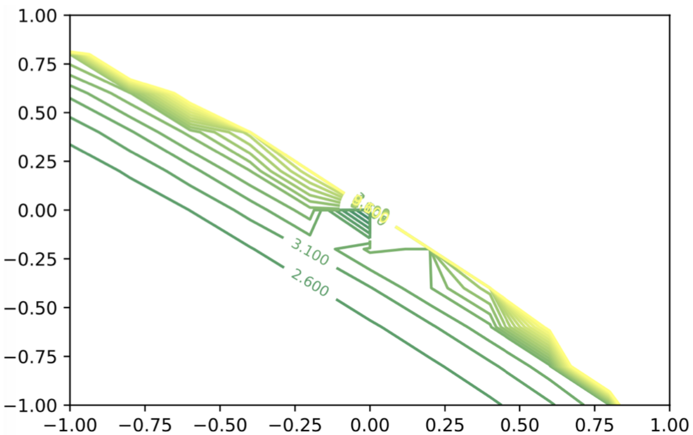
$$f(\alpha) = L(\theta^s + \alpha(\theta^l - \theta^s))$$

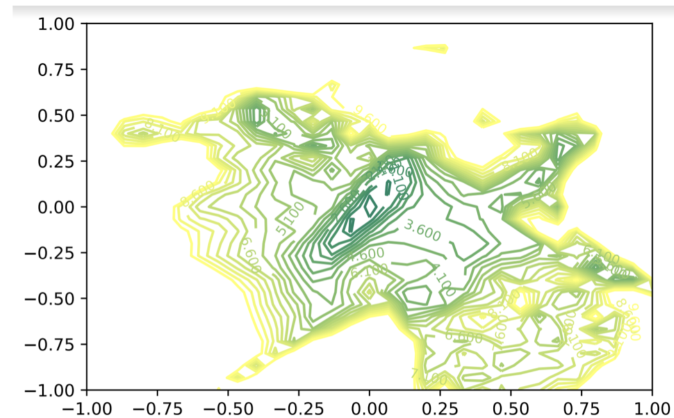
Sharpness has no relationship with generalization.

Small batch lead to better generalization.

Experimental Results & Analysis

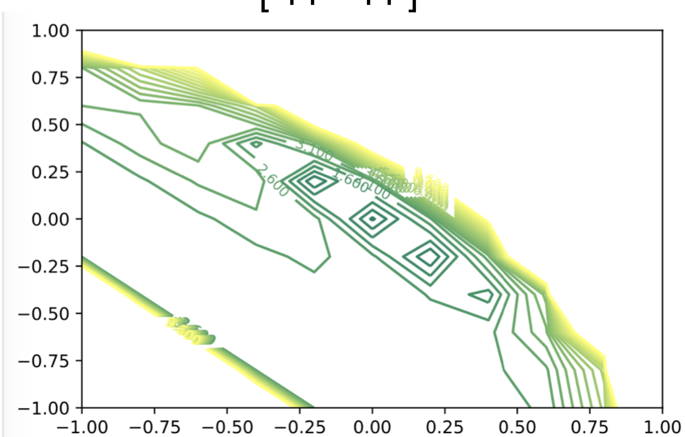
Resnet56(no shortcut), batch size=128

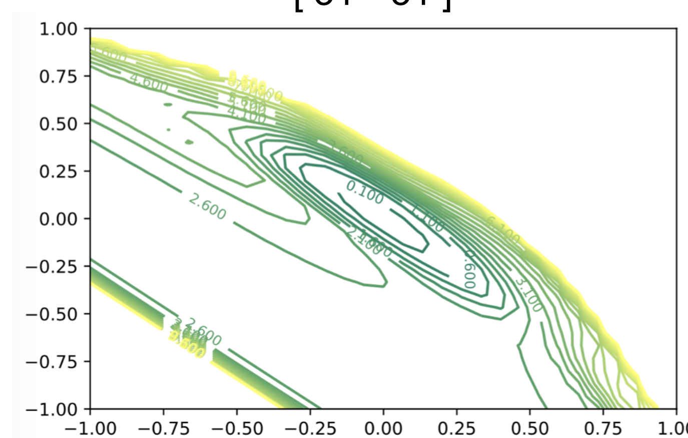




[11 * 11]

[31 * 31]





Resnet56

Conclusion and Future Work

- Filter-wise Normalization works well to show intrinsic loss landscape
- Network with smaller batch size can generalize better
  - Sharpness has no relationship with generalization
- Shortcut connections have a dramatic effect on the loss surface
  - Shortcut connections prevent the transition to chaotic behavior
- Future works:
  - Get plots on higher resolution
  - Find a simpler and faster method to do loss visualization

Job Split

Yu Du:

Load Data

1D Interpolation Graph

Training

Jupyter Notebook Wrap-up

Haochuan Zhang:

Model Data Extraction

Filter-wise Normalization

2D Contour Map

Training

References

[1] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, Tom Goldstein. Visualizing the Loss Landscape of Neural Nets.

[2] Yuanzhi Li and Yang Yuan. Convergence analysis of two-layer neural networks with relu activation. *arXiv preprint arXiv:1705.09886*, 2017.

[3] Mahdi Soltanolkotabi, Adel Javanmard, and Jason D Lee. Theoretical insights into the optimization landscape of over-parameterized shallow neural networks. *arXiv preprint arXiv:1707.04926*, 2017.

[4] Yuandong Tian. An analytical formula of population gradient for two-layered relu network and its applications in convergence and critical point analysis. In *ICML*, 2017.

[5] Ian J Goodfellow, Oriol Vinyals, and Andrew M Saxe. Qualitatively characterizing neural network optimization problems. In *ICLR*, 2015.