OPT2020: 12th Annual Workshop on Optimization for Machine Learning

On Regularization of Gradient Descent, Layer Imbalance and Flat Minima

Boris Ginsburg
NVIDIA, USA
BGINSBURG@NVIDIA.COM

Abstract

We analyze the training dynamics for deep linear networks using a new metric – layer imbalance – which defines the flatness of a solution. We demonstrate that different regularization methods, such as weight decay or noise data augmentation, behave in a similar way. Training has two distinct phases: 1) 'optimization' and 2) 'regularization'. First, during the optimization phase, the loss function monotonically decreases, and the trajectory goes toward a minima manifold. Then, during the regularization phase, the layer imbalance decreases, and the trajectory goes along the minima manifold toward a flat area. Finally, we extend the analysis for stochastic gradient descent and show that SGD works similarly to noise regularization.

1. Introduction

In this paper, we analyze regularization methods used for training of deep neural networks. To understand how regularization like weight decay and noise data augmentation work, we study gradient descent (GD) dynamics for deep linear networks (DLNs). We study deep networks with *scalar* layers to exclude factors related to over-parameterization and to focus on factors specific to deep models. Our analysis is based on the concept of *flat minima* [5]. We call a region in weight space *flat*, if each solution from that region has a similar small loss. We show that minima flatness is related to a new metric, *layer imbalance*, which measures the difference between the norm of network layers. Next, we analyze layer imbalance dynamics of gradient descent (GD) for DLNs using a *trajectory-based* approach [10]. With these tools, we prove the following results:

1. Standard regularization methods such as weight decay and noise data augmentation, decrease layer imbalance during training and drive trajectory toward flat minima.

2. Training for GD with regularization has two distinct phases: (1) 'optimization' and (2) 'regularization'. During the optimization phase, the loss monotonically decreases, and the trajectory goes toward minima manifold. During the regularization phase, layer imbalance decreases and the trajectory goes along minima manifold toward flat area.

3. Stochastic Gradient Descent (SGD) works similarly to implicit noise regularization.

2. Linear neural networks

We begin with a linear regression $y = w \cdot x + b$ with mean squared error (MSE) on N scalar samples $\{x_i, y_i\}$: $E(w, b) = \frac{1}{N} \sum (w \cdot x_i + b - y_i)^2 \rightarrow min$. The training dataset is centered and normalized:

© B. Ginsburg.

$\sum x_i = 0$; $\frac{1}{N}\sum x_i^2 = 1$; $\sum y_i = 0$; $\frac{1}{N}\sum x_i y_i = 1$. The solution for this normalized linear regression is $(w, b) = (1, 0)$.
Next, let's replace $y = w \cdot x + b$ with a linear network with d scalar layers $\boldsymbol{w} = (w_1, \ldots, w_d)$: $y = w_1 \cdots w_d \cdot x + b$. Denote $W := w_1 \cdots w_d$. The network is trained with MSE loss function: $E(\boldsymbol{w}, b) = \frac{1}{N}\sum (W \cdot x_i + b - y_i)^2 \to min$.
DLN training on the normalized dataset is equivalent to the non-convex optimization problem [2]:

$$L(\boldsymbol{w}) = (w_1 \cdots w_d - 1)^2 = (W - 1)^2 \to \min \tag{1}$$

2.1. Flat minima and Layer imbalance

Compute the loss gradient $\frac{\partial L}{\partial w_i} = 2(w_1 \cdots w_d - 1)(w_1 \cdots w_{i-1} w_{i+1} \cdots w_d) = 2(W - 1)(W/w_i)$.
Here we denote $\mathbf{W}/\mathbf{w_i} := \mathbf{w_1} \cdots \mathbf{w_{i-1}} \cdot \mathbf{w_{i+1}} \cdots \mathbf{w_d}$ for brevity.

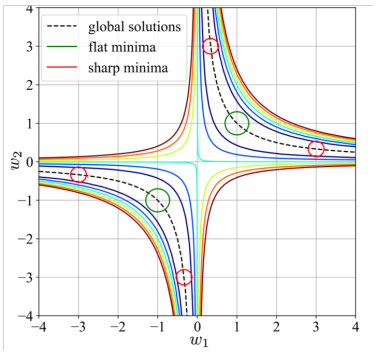


Figure 1: 2D-contour plot of the loss $L(w_1, w_2) = (w_1 w_2 - 1)^2$ for the linear network with two layers. The loss L has only global minima, located on the hyperbola $w_1 w_2 = 1$. Minima near $(-1, -1)$ and $(1, 1)$ are flat, and minima near the axes are sharp.

The minima of loss L are located on hyperbola $w_1 \cdots w_d = 1$ (see Fig. 1). Following Hochreiter et al [5, 6], we are interested in *flat minima* – "a region in weight space with the property that each weight from that region has similar small error". In contrast, sharp minima are regions where the function can increase rapidly. Hochreiter et al suggested that flat minima have smaller generalization errors than sharp minima. Keskar et al. [7] observed that large-batch training tends to converge towards a sharp minima with large positive eigenvalues of Hessian, and suggested that sharp minima generalize worse than flat minima.

In contrast, Dinh et al. [4] argued that flatness of minima can't directly applied to explain generalization; since both flat and sharp minima represent the same function, they perform equally on a validation set. Dinh showed that minima flatness is defined by the largest Hessian eigenvalue. For $L(\boldsymbol{w}) = (w_1 \cdots w_d - 1)^2 = (W - 1)^2$ the Hessian $H(\boldsymbol{w})$ is:

$$H(\boldsymbol{w}) = 2\begin{bmatrix} \frac{W^2}{w_1^2} & \frac{(2W-1)W}{w_1 w_2} & \cdots & \frac{(2W-1)W}{w_1 w_d} \\ \frac{(2W-1)W}{w_2 w_1} & \frac{W^2}{w_2^2} & \cdots & \frac{(2W-1)W}{w_2 w_d} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{(2W-1)W}{w_d w_1} & \frac{(2W-1)W}{w_d w_2} & \cdots & \frac{W^2}{w_d^2} \end{bmatrix}$$

The eigenvalues of $H(\boldsymbol{w})$ are $\{0,\dots,0,\sum\frac{1}{w_i^2}\}$. The largest eigenvalue $\big(\sum\frac{1}{w_i^2}\big)$ defines a minimizer flatness. Note that flat minima are balanced: $|w_i|\approx 1$ for all layers.

In the spirit of [1, 9], let's define *layer imbalance* for a deep linear network:

$$D(\boldsymbol{w}) := \max_{i,j} |\ ||w_i||^2 - ||w_j||^2\ | \tag{2}$$

Note that sharp minima (close to the axes) have high layer imbalance.[1]

3. The analysis of layer imbalance for gradient descent

Following Saxe et al. [10] we take a time limit for GD step: $w_i(t+1)=w_i(t)-\lambda\cdot\nabla L(\boldsymbol{w})$, to obtain *Continuous Gradient Descent* CGD: $\frac{dw_i}{dt}=-\lambda\frac{\partial L}{\partial w_i}=-2\lambda(W-1)(W/w_i)$.
For CGD, the loss L monotonically decreases:

$$\frac{dL}{dt}=\sum\big(\frac{\partial L}{\partial w_i}\cdot\frac{dw_i}{dt}\big)=-4\lambda(W-1)^2W^2\big(\sum\frac{1}{w_i^2}\big)=-4\lambda W^2\big(\sum\frac{1}{w_i^2}\big)\cdot L(t)\le 0$$

and the CGD trajectory is a hyperbola: $w_i^2(t)-w_j^2(t)=\text{const}$ (see Fig. 2a) [10]. The layer imbalance remains constant during training. If training starts close to the origin, then a final point will also have a small layer imbalance and a minimum will be flat.

Let's turn from CGD back to the regular GD[2]: $w_i(t+1)=w_i-2\lambda\frac{\partial L}{\partial w_i}=w_i-2\lambda(W-1)(W/w_i)$. We would like to find conditions, which would guarantee that the loss monotonically decreases. Note that for any *fixed* learning rate, one can find a point $\boldsymbol{w}$, such that the loss will increase after the GD step. But we can define an *adaptive* learning rate $\lambda(\boldsymbol{w})$ which guarantees that the loss decreases.

Theorem 1 *Consider GD:* $w_i(t+1)=w_i-2\lambda(W-1)(W/w_i)$. *Assume that* $|W-1|<\frac{1}{2}$. *If we define an adaptive learning rate* $\lambda(\boldsymbol{w})=\frac{1}{4\sum(1/w_i^2)}$, *then the loss monotonically converges to 0 with a linear rate, and the layer imbalance monotonically decreases.*

Proof The convergence rate is proved in App. A.1. The layer imbalance analysis is in App. A.2. ■

Note that we proved only that the layer imbalance D decreases, but not that D converges to 0. The layer imbalance may stay large, if the loss $L\to 0$ too fast or if $W\approx 0$, so the factor $k=1-4\lambda^2\cdot L\cdot W^2(1/(w_iw_j))^2\to 1$. To make the layer imbalance $D\to 0$, we should keep the loss in certain range, e.g. $\frac{1}{4}<|W-1|<\frac{1}{2}$. For this, we could increase the learning rate if the loss becomes too small, and decrease learning rate if loss becomes large.

4. The analysis of layer imbalance for gradient descent with explicit regularization

In this section, we prove that the layer imbalance decreases for GD with explicit regularization, such as weight decay, noise data augmentation, dropout etc.

1. The question of how minima flatness is related to generalization is out of scope of this paper. Our interest in flat minima is related to training robustness. Gradient descent is more stable in the flat area than in the sharp area: the gradient $\frac{\partial L}{\partial w_i}$ vanishes if $|w_i|$ is very large, and the gradient explodes if $|w_i|$ is very small.
2. We omit t in the right part for brevity, so w_i means $w_i(t)$.

4.1. The analysis of layer imbalance for gradient descent with weight decay

As before, we consider the GD for linear network $(w_1, \dots, w_d)$ with d layers. Let's add the weight decay (WD) term to the loss: $\bar{L}(\boldsymbol{w}) = (w_1 \cdots w_d - 1)^2 + \mu(w_1^2 + \cdots + w_d^2)$.
The CGD with weight decay is described by the following DEs:

$$\frac{dw_i}{dt} = -\lambda \frac{\partial \bar{L}}{\partial w_i} = -2\lambda\big((W-1)(W/w_i) + \mu \cdot w_i\big) \tag{3}$$

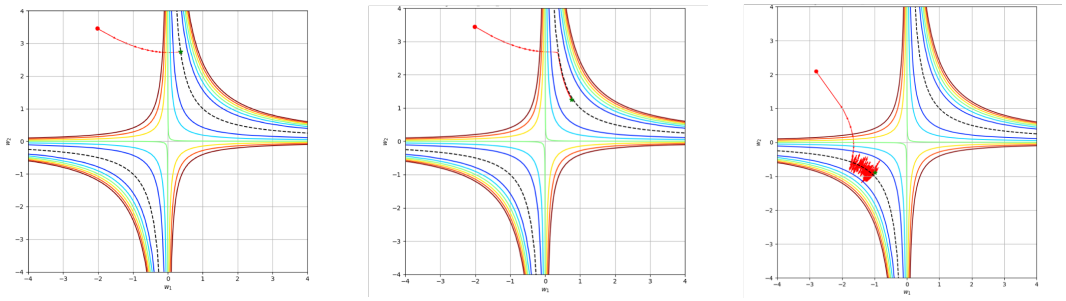
Accordingly, the loss dynamics for CGD with weight decay is:

$$\frac{dL}{dt} = \sum \frac{\partial L}{\partial w_i} \cdot \frac{dw_i}{dt} = -4\lambda\Big((W-1)^2 W^2 \big(\sum 1/w_i^2\big) + \mu \cdot d \cdot (W-1)W\Big)$$
$$= -4\lambda\big(\sum 1/w_i^2\big) W^2 (W-1)\big(W - (1 - \mu \frac{d}{W(\sum 1/w_i^2)})\big)$$

The loss decreases when $k = (W-1)\big(W - (1 - \mu \frac{d}{W(\sum 1/w_i^2)})\big) > 0$, outside the *weight decay band*: $1 - \mu \frac{d}{W(\sum 1/w_i^2)} \le W \le 1$. The width of this band is controlled by the weight decay μ.

We can divide GD training with weight decay into two phases: (1) optimization and (2) regularization. During the first phase, the loss decreases until the trajectory gets into the WD-band. During the second phase, the loss L can oscillate, but the trajectory stays inside the WD-band (Fig. 2b) and goes toward a flat minima area. The layer imbalance monotonically decreases during second phase:

$$\frac{d(w_i^2 - w_j^2)}{dt} = -4\lambda \cdot \Big(\big((W-1)W + \mu w_i^2\big) - \big((W-1)W + \mu w_j^2\big)\Big) = -4\lambda \cdot \mu \cdot (w_i^2 - w_j^2)$$



(a) Continuous GD (b) GD with weight decay (c) GD with noise augmentation

Figure 2: The training trajectories for (a) continuous GD, (b) GD with weight decay, and (c) GD with noise augmentation. The trajectory for continuous GD is a hyperbola: $w_i^2(t) - w_j^2(t) = \text{const}$. The trajectories for GD with weight decay and noise augmentation have two parts: (1) optimization – the trajectory goes toward the minima manifold, and (2) regularization – the trajectory goes along minima manifold toward a flat area.

4.2. The analysis of layer imbalance for gradient descent with noise augmentation

Bishop [3] showed that for shallow networks, training with noise is equivalent to Tikhonov regularization. We will extend this result to DLNs. Let's augment the training data with noise: $\tilde{x} = x \cdot (1+\eta)$, where the noise η has $0-$mean and is bounded: $|\eta| \leq \delta < \frac{1}{2}$. The DLN with noise augmentation can be written in the following form:

$$\tilde{y} = w_1 \cdots w_d \cdot (1+\eta)x = W \cdot (1+\eta)x. \tag{4}$$

This model also describes continuous dropout [11] when layer outputs h_i are multiplied with the noise: $\tilde{h}_i = (1+\eta) \cdot h_i$. This model can be also used for continuous drop-connect [8, 12] when the noise is applied to weights: $\tilde{w}_i = (1+\eta) \cdot w_i$. The CGD with noise augmentation is described by the following DEs: $\frac{dw_i}{dt} = -\lambda \frac{\partial \tilde{L}}{\partial w_i} = -2\lambda \cdot (1+\eta)(W(1+\eta)-1)(W/w_i)$.

Let's consider loss dynamics:

$$\frac{dL}{dt} = \sum \Big(\frac{\partial L}{\partial w_i} \cdot \frac{dw_i}{dt} \Big) = -4\lambda(1+\eta)W^2 \big(\sum 1/w_i^2 \big)(W-1)(W(1+\eta)-1)$$
$$= -4\lambda(1+\eta)^2 W^2 \big(\sum 1/w_i^2 \big) \cdot \Big((W-1)(W - \frac{1}{1+\eta}) \Big)$$

The loss decreases while the factor $k = (W-1)(W - \frac{1}{1+\eta}) = (W-1)(W-1-\frac{\eta}{1+\eta}) > 0$, outside of the *noise band* $1 - \frac{\delta}{1+\delta} < W < 1 + \frac{\delta}{1-\delta}$. The training trajectory is the hyperbola $w_i^2(t) - w_j^2(t) = \text{const}$. When the trajectory gets inside the noise band, it oscillates around the minima manifold, but the layer imbalance remains constant for continuous GD.

For discrete GD, noise augmentation works similarly to weight decay. Training has two phases: (1) optimization and (2) regularization (Fig. 2c). During the first phase, the loss decreases until the trajectory hits the noise band. Next, the trajectory oscillates inside the noise band, and the layer imbalance decreases. The noise variance σ^2 defines the band width, similarly to the weight decay μ.

Theorem 2 *Consider GD with noise* $w_i(t+1) = w_i - 2\lambda(1+\eta)(W(1+\eta)-1)(W/w_i)$. *Assume that the noise* η *has 0-mean and it is bounded:* $|\eta| < \delta < \frac{1}{2}$. *If we define the adaptive learning rate* $\lambda(\boldsymbol{w}) = \lambda_0 \frac{1}{\sum 1/w_i^2}$, *then the layer imbalance monotonically decreases inside the noise band* $|W-1| < \delta$. *The layer imbalance monotonically converges to* 0, *if layers are also uniformly bounded:* $|w_i| < C$.

Proof See Appendix. A.3. ■

5. The analysis of layer imbalance for SGD

In this section, we show that the layer imbalance converges to 0 for SGD, and that SGD works as implicit noise regularization. As before, we train a linear network $y = Wx$ with loss $L(\boldsymbol{w}) = \frac{1}{N}\sum (Wx_n - y_n)^2$. The dataset $\{x_n, y_n\}$ is normalized: $\sum x_i = 0$; $\frac{1}{N}\sum x_i^2 = 1$; $\sum y_i = 0$; $\frac{1}{N}\sum x_i y_i = 1$. A stochastic gradient for a batch $\bar{B}$ with $B < N$ samples is:

$$\frac{\partial L_B}{\partial w_i} = \frac{1}{|B|} \sum_{\bar{B}} 2(Wx_n^2 - x_n y_n)W/w_i = 2\Big(W(\frac{1}{B}\sum_{\bar{B}} x_n^2) - (\frac{1}{B}\sum_{\bar{B}} x_n y_n) \Big) W/w_i$$

If batch size $B \to N$, then terms $\sum_{\bar{B}} x_n^2 \to \sum_N x_n^2 = 1$ and $\sum_{\bar{B}}(x_n y_n) \to \sum_{\bar{B}}(x_n y_n) = 1$. So we can write the stochastic gradient in the following form:

$$\frac{\partial L_B}{\partial w_i} = 2\Big(W(1+\eta_1) - (1+\eta_2)\Big)W/w_i = 2\Big(W - 1 + (W\eta_1 - \eta_2)\Big)W/w_i$$

The factor $(1+\eta_1)$ works as noise data augmentation, and the term η_2 works as label noise. Both η_1 and η_2 have 0-mean. When loss is small, we can combine both components into one *SGD noise* term: $\eta = W\eta_1 - \eta_2$. SGD noise η has 0-mean. We assume that SGD noise variance depends on batch size in the following way: $\sigma^2 \approx (\frac{1}{B} - \frac{1}{N})$.
The trajectory for continuous SGD is described by the stochastic DEs:

$$\frac{dw_i}{dt} = -\lambda \cdot \frac{\partial L_B}{\partial w_i} = -2\lambda\Big(W - 1 + \eta\Big)W/w_i$$

Let's start with loss analysis: $\frac{dL}{dt} = -4\lambda W^2\big(\sum 1/w_i^2\big) \cdot (W-1)(W-1+\eta)$. For continuous SGD, the loss decreases anywhere except in the *SGD noise band*: $(W-1)(W-1+\eta) < 0$. The band width depends on B: the smaller the batch, the wider the band. The SGD training consists of two parts. First, the loss decreases until the trajectory hits the SGD-noise band. Then the trajectory oscillates inside the noise band. The layer imbalance remains constant for continuous SGD. But the layer imbalance decreases for discrete SGD.

Theorem 3 *Consider SGD:* $w_i(t+1) = w_i - \lambda \cdot \frac{\partial L_B}{\partial w_i}$. *Assume that* $|W-1| < \delta$, *and that SGD noise satisfies* $|\eta| \le \delta < 1$. *Then the layer imbalance monotonically decreases for the adaptive learning rate* $\lambda(\boldsymbol{w}) = \frac{1}{2\delta(1+\delta)(\sum(1/w_i^2)}$.

Proof See Appendix. A.4 ■

The layer imbalance $D \to 0$ at a rate proportional to the variance of SGD noise. It was observed by Keskar et al. [7] that SGD training with a large batch leads to sharp solutions, which generalize worse than solutions obtained with a smaller batch. This fact directly follows from Theorem 3. The layer imbalance decreases at a rate $O(1 - k\lambda^2\sigma^2)$. When a batch size increases, $B \to N$, the variance of SGD-noise decreases as $\approx (\frac{1}{B} - \frac{1}{N})$. One can compensate for smaller SGD noise with additional generalization: data augmentation, weight decay, dropout, etc.

6. Discussion

In this work, we explore dynamics for gradient descent training of deep linear networks. Using the layer imbalance metric, we analyze how regularization methods such as weight decay, data augmentation, dropout, etc, affect training dynamics. We show that for all these methods the training has two distinct phases: optimization and regularization. During the optimization phase, the training trajectory goes from an initial point toward minima manifold, and loss monotonically decreases. During the regularization phase, the trajectory goes along minima manifold toward flat minima, and the layer imbalance monotonically decreases. We showed that noise augmentation and continuous dropout work similarly to L_2-regularization. Finally, we show that SGD behaves in the same way

as gradient descent with noise regularization. This work provides an analysis of regularization for scalar linear networks. We leave the question of how regularization works for over-parameterized nonlinear networks for future research.

References

[1] S. Arora, N. Golowich, N. Cohen, and W. Hu. A convergence analysis of gradient descent for deep linear neural networks. In *ICLR*, 2019.

[2] P. Baldi and K. Hornik. Neural networks and principal component analysis: Learning from examples without local minima. In *Neural Networks 2.1*, page 53–58, 1989.

[3] C. M. Bishop. Training with noise is equivalent to Tikhonov regularization. *Neural Computation*, 7:108–116., 1995.

[4] L. Dinh, R. Pascanu, S. Bengio, and Y. Bengio. Sharp minima can generalize for deep nets. In *ICML*, 2017.

[5] S. Hochreiter and J. Schmidhuber. Simplifying neural nets by discovering flat minima. In *NIPS*, 1994.

[6] S. Hochreiter and J. Schmidhuber. Flat minima search for discovering simple nets, technical report fki-200-94. Technical report, Fakultat fur Informatik, H2, Technische Universitat Munchen, 1994.

[7] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang. On large-batch training for deep learning: generalization gap and sharp minima. In *ICLR*, 2016.

[8] D. Kingma, T. Salimans, and M. Welling. Variational dropout and the local reparameterization trick. In *NIPS*, 2015.

[9] B. Neyshabur, R. Salakhutdinov, and N. Srebro. Path-SGD: Path-normalized optimization in deep neural networks. In *NIPS*, 2015.

[10] A. M. Saxe, J. L. McClelland, and S. Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural network. In *ICLR*, 2013.

[11] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 2014.

[12] L. Wan, M. Zeiler, S. Zhang, Y. LeCun, and R. Fergus. Regularization of neural networks using dropconnect. In *ICML*, 2013.

Appendix A. Proofs

A.1. Convergence analysis for gradient descent

Theorem 4 *Consider discrete GD:*

$$w_i(t+1) = w_i - 2\lambda \frac{\partial L}{\partial w_i} = w_i - 2\lambda(W-1)(W/w_i)$$

Assume that $|W-1| < \frac{1}{2}$. *If we define an adaptive learning rate* $\lambda(\boldsymbol{w}) = \frac{1}{4\sum(1/w_i^2)}$, *then the loss monotonically converges to 0 with a linear rate.*

Proof Let's estimate the loss change for a GD step:

$$\begin{aligned}
&W(t+1) - 1 = \prod \big(w_i - 2\lambda(W-1)W/w_i\big) - 1 \\
&= \prod \big(w_i(1 - 2\lambda(W-1)W/w_i^2)\big) - 1 = W \cdot \prod \big(1 - 2\lambda(W-1)W/w_i^2\big) - 1 \\
&= W \cdot \Big(1 - 2\lambda(W-1)W\big(\sum_i 1/w_i^2\big) + 4\lambda^2(W-1)^2W^2\big(\sum_{i\neq j} 1/(w_i^2w_j^2)\big) \\
&\quad - 8\lambda^3(W-1)^3W^3\big(\sum_{i\neq j\neq k} 1/(w_i^2w_j^2w_k^2)\big) + ...\Big) - 1 \\
&= (W-1) \cdot \Big(1 - 2\lambda W^2\big(\sum_i 1/w_i^2\big) + 4\lambda^2(W-1)W^3\big(\sum_{i\neq j} 1/(w_i^2w_j^2)\big) \\
&\quad - 8\lambda^3(W-1)^2W^4\big(\sum_{i\neq j\neq k} 1/(w_i^2w_j^2w_k^2)\big) + ...\Big) = (W-1) \cdot \Big(1 - \frac{W}{W-1} \cdot S\Big)
\end{aligned}$$

Here $S = a_1 - a_2 + a_3 - ... + a_d$ is a series with $a_k = \big(2\lambda(W-1)W\big)^k\big(\sum_{i\neq j\neq ...m} 1/(w_i^2w_j^2...w_m^2)\big)$:

$$\begin{aligned}
S = 2\lambda(W-1)W\big(\sum_i 1/w_i^2\big) - 4\lambda^2(W-1)^2W^2\big(\sum_{i\neq j} 1/(w_i^2w_j^2)\big) \\
+ 8\lambda^3(W-1)^3W^3\big(\sum_{i\neq j\neq k} 1/(w_i^2w_j^2w_k^2)\big) + \dots
\end{aligned}$$

Consider the factor $k = \big(1 - \frac{W}{W-1} \cdot S\big)$. To prove that $|k| < 1$, we consider two cases.

CASE 1: $(W-1)W < 0$. In this case, the series S can be written as:

$$\begin{aligned}
S = -\Big(2\lambda(1-W)W\big(\sum_i 1/w_i^2\big) + 4\lambda^2(1-W)^2W^2\big(\sum_{i\neq j} 1/(w_i^2w_j^2)\big) + \\
+ 8\lambda^3(1-W)^3W^3\big(\sum_{i\neq j\neq k} 1/(w_i^2w_j^2w_k^2)\big) + ...\Big) \geq 2\lambda(W-1)W\big(\sum_i 1/w_i^2\big)\frac{1}{1-q}
\end{aligned}$$

where q is:

$$\begin{aligned}
q = \Big|\frac{a_{k+1}}{a_k}\Big| = \left|\frac{(2\lambda(W-1)W)^{k+1}\big(\sum_{i\neq ...\neq m+1} 1/(w_i^2...w_{m+1}^2)\big)}{(2\lambda\,(W-1)W)^k\big(\sum_{i\neq ...\neq m} 1/(w_i^2...w_m^2)\big)}\right| \\
\leq 2\lambda|(W-1)W|\frac{\big(\sum_{i\neq ...\neq m} 1/(w_i^2...w_m^2)\big)\big(\sum 1/w_i^2\big)}{\sum_{i\neq ...\neq m} 1/(w_i^2...w_m^2)} = 2\lambda|(W-1)W|\big(\sum 1/w_i^2\big) \leq \frac{3}{8}
\end{aligned}$$

So on the one hand: $k = 1 - \frac{W}{W-1}S \geq 1 - \frac{W}{W-1} \cdot 2\lambda(W-1)W(\sum 1/w_i^2)\frac{1}{1-q} \geq -\frac{4}{5}$.

On the other hand: $k < 1 - \frac{W}{W-1} \cdot 2\lambda(W-1)W(\sum_i 1/w_i^2) = 1 - 2\lambda W^2(\sum 1/w_i^2) < \frac{7}{8}$.

CASE 2: $(W-1)W > 0$. In the series $S = a_1 - a_2 + a_3 - ...$, all terms a_i are now positive. Since $q = \left|\frac{a_{k+1}}{a_k}\right| < \frac{3}{8}$, we have that $\frac{5}{8}a_1 < a_1 - a_2 < S < a_1$.

On the one hand: $k = 1 - \frac{W}{W-1}S \geq 1 - \frac{W}{W-1}a_1 = 1 - 2\lambda(\sum 1/w_i^2) \cdot W^2 > -\frac{1}{8}$.

On the other hand: $k = 1 - \frac{W}{W-1}S \leq 1 - \frac{5}{8} \cdot \frac{W}{W-1}a_1 = 1 - \frac{5}{8} \cdot 2\lambda(\sum 1/w_i^2) \cdot W^2 < \frac{59}{64}$.

To conclude, in CASE 1 we prove that $-\frac{4}{5} < k < \frac{7}{8}$ and in CASE 2 that $-\frac{1}{8} < k < \frac{59}{64}$.

Since $L(t+1) < L(t) \cdot k^2$, the loss L monotonically converges to 0 with rate k^2. ■

A.2. Implicit regularization for discrete gradient descent

Theorem 5 *Consider discrete GD*

$$w_i(t+1) = w_i - 2\lambda\frac{\partial L}{\partial w_i} = w_i - 2\lambda(W-1)(W/w_i)$$

Assume that $|W-1| < \frac{1}{2}$. If we define an adaptive learning rate $\lambda(\boldsymbol{w}) = \frac{1}{4\sum(1/w_i^2)}$, then the layer imbalance monotonically decreases.

Proof Let's compute the layer imbalance D_{ij} for the layers i and j after one GD step:

$$D_{ij}(t+1) = w_i(t+1)^2 - w_j(t+1)^2 = \left(w_i - 2\lambda(W-1)W/w_i\right)^2 - \left(w_j - 2\lambda(W-1)W/w_j\right)^2$$
$$= (w_i^2 - w_j^2) \cdot \left(1 - 4\lambda^2(W-1)^2W^2/(w_iw_j)^2\right) = D_{ij} \cdot \left(1 - 4\lambda^2(W-1)^2W^2/(w_iw_j)^2\right)$$

On the one hand, the factor $k = 1 - 4\lambda^2(W-1)^2W^2/(w_iw_j)^2 \leq 1$.

On the other hand:

$$k = 1 - 4\lambda^2(W-1)^2W^2/(w_iw_j)^2 \geq 1 - \lambda^2(W-1)^2W^2(1/w_i^2 + 1/w_j^2)^2$$
$$\geq 1 - \lambda^2(\sum 1/w_l^2)^2(W-1)^2W^2 \geq 1 - \frac{9}{256} = \frac{247}{256}$$

So $D_{ij}(t+1) = k \cdot D_{ij}(t)$ and $\frac{247}{256} < k \leq 1$. This guarantees that the layer imbalance decreases. ■

A.3. Training with noise augmentation

Theorem 6 *Consider discrete GD with noise augmentation*

$$w_i(t+1) = w_i - 2\lambda(1+\eta)(W(1+\eta) - 1)(W/w_i)$$

Assume that the noise η has 0-mean and is bounded: $|\eta| < \delta < \frac{1}{2}$. If we define the adaptive learning rate $\lambda(\boldsymbol{w}) = \frac{1}{2}\left(\frac{2}{3}\right)^5 \frac{1}{\sum 1/w_i^2}$, then the layer imbalance monotonically decreases inside the noise band $|W-1| < \delta$.

Proof Let's estimate the layer imbalance:

$$\begin{aligned}
&w_i^2(t+1) - w_j^2(t+1) \\
&= \big(w_i - 2\lambda(1+\eta)(W(1+\eta)-1)W/w_i\big)^2 - \big(w_j - 2\lambda(1+\eta)(W(1+\eta)-1)W/w_j\big)^2 \\
&= (w_i^2 - w_j^2) + 4\lambda^2(1+\eta)^2(W(1+\eta)-1)^2\big(W^2/w_i^2 - W^2/w_j^2\big) \\
&= (w_i^2 - w_j^2)\cdot\Big(1 - 4\lambda^2(1+\eta)^4\big(W - \frac{1}{1+\eta}\big)^2 W^2/(w_i w_j)^2\Big)
\end{aligned}$$

On the one hand, the factor $k = 1 - 4\lambda^2(1+\eta)^4\big(W - \frac{1}{1+\eta}\big)^2 W^2/(w_i w_j)^2 \le 1$.

On the other hand:

$$\begin{aligned}
k &= 1 - 4\lambda^2(1+\eta)^4\big(W - \frac{1}{1+\eta}\big)^2 W^2/(w_i w_j)^2 \\
&\ge 1 - \lambda^2(1+\eta)^4\big(W - \frac{1}{1+\eta}\big)^2 W^2(1/w_i^2 + 1/w_j^2)^2 \\
&\ge 1 - \lambda^2(1+\eta)^4\big(W - 1 + \frac{\eta}{1+\eta}\big)^2 W^2\big(\sum_i 1/w_i^2\big)^2 \\
&\ge 1 - \lambda^2\big(\sum_i 1/w_i^2\big)^2\cdot(1+\delta)^4\big(\delta + \frac{\delta}{1-\delta}\big)^2(1+\delta)^2 \ge 1 - \lambda^2\big(\sum_i 1/w_i^2\big)^2(3/2)^{10}
\end{aligned}$$

Taking $\lambda = \frac{1}{2}\big(\frac{2}{3}\big)^5\frac{1}{\sum 1/w_i^2}$ makes $0 < k \le 1$, which proves that the layer imbalance decreases. ■

We can prove that the layer imbalance $E[D] \to 0$ if we also assume that all layers are uniformly bounded $|w_i| < C$. This implies that there is $\epsilon > 0$ such that for all $\boldsymbol{w}$ the adaptive learning rate $\lambda(\boldsymbol{w}) > \epsilon$, and we can prove that the expectation $E(k) < 1$:

$$\begin{aligned}
E(k) &= 1 - E\Big[4\lambda^2(1+\eta)^4\big(W - \frac{1}{1+\eta}\big)^2 W^2/(w_i w_j)^2\Big] \\
&\le 1 - 4\lambda^2 W^2/(w_i w_j)^2\cdot(1+\sigma^2)^2\frac{\sigma^2}{1+\sigma^2} \le 1 - 4\lambda^2\frac{1}{4C^4}\big(1+\sigma^2\big)\sigma^2 \le 1 - \frac{\lambda^2\sigma^2}{C^4}
\end{aligned}$$

This proves that the layer imbalance $D \to 0$ with rate $\big(1 - \frac{\lambda^2\sigma^2}{C^4}\big)$.

A.4. SGD noise as implicit regularization

Theorem 7 *Consider discrete SGD*

$$w_i(t+1) = w_i - 2\lambda(W - 1 + \eta)W/w_i$$

Assume that $|W-1| < \delta$, and that SGD noise satisfies $|\eta| \le \delta < 1$. If we define the adaptive learning rate $\lambda(\boldsymbol{w}) = \frac{1}{2\delta(1+\delta)(\sum(1/w_i^2)}$, then the layer imbalance monotonically decreases.

Proof Let's estimate the layer imbalance:

$$w_i^2(t+1) - w_j^2(t+1) = \big(w_i - 2\lambda(W-1+\eta)W/w_i\big)^2 - \big(w_j - 2\lambda(W-1+\eta)W/w_j\big)^2$$
$$= (w_i^2 - w_j^2) \cdot \Big(1 - 4\lambda^2(W-1+\eta)^2W^2/(w_iw_j)^2\Big)$$

On the one hand, the factor $k = 1 - 4\lambda^2(W-1+\eta)^2W^2/(w_iw_j)^2 \leq 1$. On the other hand:

$$k = 1 - 4\lambda^2(W-1+\eta)^2W^2/(w_iw_j)^2 \geq 1 - 2\lambda^2(W-1+\eta)^2W^2\big(1/w_i^2 + 1/w_j^2\big)^2$$
$$\geq 1 - 4\lambda^2W^2\big(\sum 1/w_i^2\big)^2 \cdot ((W-1)^2 + \eta^2) \geq 1 - 4\lambda^2\big(\sum 1/w_i^2\big)^2 \cdot \delta^2(1+\delta)^2$$

Setting $\lambda = \dfrac{1}{2\delta(1+\delta)(\sum(1/w_i^2)}$ makes $0 < k \leq 1$, which completes the proof. ■