Exponential Convergence Time of Gradient Descent for One-Dimensional Deep Linear Neural Networks

By Ohad Shamir

Ali Zindari
M.Sc. in Mathematics & Computer Science at Saarland University

Saarbrücken, Germany

January 22, 2025

Deep Linear Models

- Deep linear models are simplified versions of multilayer perceptrons (MLPs).
- Objective: Minimize

$$f\left(\prod_{i=1}^{L} W_i\right)$$

- If f is convex, all minima are global. Example:

$$\min_{W_1,\ldots,W_L} \left\| \prod_{i=1}^{L} W_i X - Y \right\|_F^2$$

- Key aspects of MLPs are omitted, such as activation functions, skip connections, and normalization.

Why Do More Layers Hurt? (Do They?)

Let's assume all weights are scalars.

- Gradient descent update:

$$w_i(t+1) = w_i(t) - \eta f' \left(\prod_{i=1}^{L} w_i(t) \right) \prod_{j \neq i} w_j(t)$$

- Xavier initialization: $w_i(0) \sim \mathcal{N}(0, 1)$. Initially, weights are typically less than 1.

Vanishing Gradients

$\prod_{j \neq i} w_j(0)$ shrinks exponentially with the number of layers! Gradients become extremely small!

Why Study Deep Linear Models?

Suppose we analyze a neural network with nonlinear activations but are only interested in the effect of depth on optimization. If adding more layers leads to faster training, can we determine whether:

- The network benefits from better feature learning? Or
- There is a genuine acceleration due to increased depth?

$$\textbf{Acceleration in Training} \begin{cases} \text{Improved feature learning,} \\ \text{Inherent acceleration from depth.} \end{cases}$$

Key Insight

Adding more layers should NOT make the network inherently more expressive!

Deep Linear Models isolate this effect perfectly!

Key Research Questions

- Does Gradient Descent (GD) converge on deep linear models?
- How fast does GD converge?
- How does depth impact convergence?
- How does initialization affect convergence?
- What crucial aspects are missing in deep linear models?

Our Approach

We first examine results from two prior works to evaluate this paper!

Findings from [Arora et al., 2018]

Problem Formulation:

$$\min_{W_1,\ldots,W_k} \left\| \prod_{i=1}^{L} W_i - Y \right\|_F^2$$

Key Assumption

Initial weights must satisfy:

$$\left\| W_{j+1}^\top W_{j+1} - W_j W_j^\top \right\|_F^2 \leq \delta \tag{1}$$

- If $\delta = 0$, all initial weight matrices must share the same singular values!
- Assumption: $\delta = \mathcal{O}(\frac{1}{L^3})$, which approaches zero rapidly.

This is unrealistic!

Convergence Guarantee

Theorem

If the initial weights satisfy the δ-assumption with $\delta = \mathcal{O}\left(\frac{1}{L^3}\right)$, the number of iterations required to reach ϵ accuracy is:

$$\mathcal{O}\left(L^3 \cdot \log\left(\frac{1}{\epsilon}\right)\right). \tag{2}$$

- The convergence rate is linear.
- The dependence on depth worsens cubically.
- The initialization assumption is overly restrictive.

Findings from [Bartlett et al., 2018]

Problem:

$$\min_{W_1,\ldots,W_L} \left\| \prod_{i=1}^{L} W_i - Y \right\|_F^2$$

Assumptions

- All weights must be initialized exactly as identity matrices.
- The target matrix Y must be positive semi-definite.

Strong assumptions!

Convergence Guarantee

Theorem

If the initial weights are identity matrices and Y is positive semi-definite, the number of iterations required to reach ϵ accuracy is:

$$\mathcal{O}\left(L \cdot \log\left(\frac{1}{\epsilon}\right)\right). \tag{3}$$

- The convergence rate is linear.
- It scales linearly with the number of layers.

Huge improvement over the previous paper, but under which conditions?

Current Paper

Problem:

$$f\left(\prod_{i=1}^{L} w_i\right)$$

- Weights are scalars.
- f is convex.

Assumptions

Weights are initialized according to Xavier initialization.

Xavier Initialization for Deep Linear Models

- Xavier initialization sets each entry of W_i as:

$$(W_i)_{m,n} \sim \mathcal{N}\left(0, \frac{1}{d}\right) \quad \text{or} \quad (W_i)_{m,n} \sim U\left(-\frac{1}{\sqrt{d}}, \frac{1}{\sqrt{d}}\right),$$

  where d is the dimensionality (width of the network).
- It is influenced by width, but not by depth.
- In [Arora et al., 2018], the initialization depends on depth.
- Xavier initialization for scalars: $w_i \sim \mathcal{N}(0, 1)$.

We cannot control the singular values as a function of depth!

Main Results

Negative Result

Gradient descent (GD) requires at least $\exp(\Omega(L)) \cdot \log(1/\epsilon)$ steps to reach ϵ accuracy under Xavier initialization.

Positive Result

With near-identity initialization, GD requires at most $\exp(\mathcal{O}(L)) \cdot \log(1/\epsilon)$ steps.

Conclusion

Overparameterization exponentially slows down GD, unlike in practical deep networks.

Different results arise due to different initializations!

Strengths and Limitations

Strengths:

- **General initialization:** Xavier initialization, commonly used in practice.
- **General results:** Hold for any convex function f.

Limitations:

- **Limited setting:** Focuses on scalar models, not matrix weights.
- **Overly pessimistic analysis:** Ignores key properties of practical deep networks.
- **Limited experiments:** Only simple linear models, no practical deep networks.

One Question

Is it possible to have a rate of $\mathcal{O}(\log(1/\epsilon))$?

Meaning that depth does not affect the convergence? Yes!

Width Matters: [Du and Hu, 2019]

Objective:

$$\min_{W_1,\ldots,W_L} \frac{1}{2} \|W_L \ldots W_1 X - Y\|_F^2$$

- $W_1 \in \mathbb{R}^{m \times d_{in}}$ and $W_2, \ldots, W_{L-1} \in \mathbb{R}^{m \times m}$, $W_L \in \mathbb{R}^{m \times d_{out}}$.
- $r = \text{rank}(X)$.
- $\kappa = \frac{\lambda_{\max}(X^\top X)}{\lambda_r(X^\top X)}$, where $\lambda_r(A)$ is the r-th largest eigenvalue of A.
- Xavier initialization is assumed.

Main Result

Theorem

Given a deep linear model with a width of $\tilde{\Omega}(Lrd_{out}\kappa^3)$, GD finds an ϵ-accuracy solution in $\mathcal{O}(\kappa \log(1/\epsilon))$ iterations.

- Depth is removed from the rate!
- Width must grow linearly with depth.
- The lower bound from [Shamir, 2019] is broken.

Residual Connection: [Zou et al., 2020]

Objective:

$$\min_{W_1,\ldots,W_L} \frac{1}{2} \|W_L(W_{L-1}+\mathbf{I})\ldots(W_2+\mathbf{I})W_1X - Y\|_F^2$$

- Residual connections: One of the most influential techniques in deep learning.
- $r = \text{rank}(X)$.
- $\kappa = \frac{\lambda_{\max}(X^\top X)}{\lambda_r(X^\top X)}$, where $\lambda_r(A)$ is the r-th largest eigenvalue of A.
- Zero initialization is assumed.

Main Result

Theorem

Given a deep linear model with a width of $\tilde{\Omega}(rd_{out}\kappa^2)$, GD finds an ϵ-accuracy solution in $\mathcal{O}(\kappa \log(1/\epsilon))$ iterations.

- Depth is removed from the rate!
- Width is independent of depth.
- The lower bound from [Shamir, 2019] is broken.
- The rate is improved over [Du and Hu, 2019] by a factor of $\mathcal{O}(L\kappa)$.

Summary of Results

Results from oldest to newest paper.

Paper	Initialization	Convergence Rate
[Arora et al., 2018]	Nearly equal singular values (δ-assumption, $\delta = \mathcal{O}(1/L^3)$)	$\mathcal{O}(L^3 \log(1/\epsilon))$
[Bartlett et al., 2018]	Identity initialization	$\mathcal{O}(L \log(1/\epsilon))$
Current Paper (LB)	Xavier initialization ($w_i \sim \mathcal{N}(0,1)$)	$\exp(\Omega(L)) \log(1/\epsilon)$
Current Paper (UB)	Near-identity initialization	$\exp(\mathcal{O}(L)) \log(1/\epsilon)$
[Du and Hu, 2019]	Xavier initialization	$\mathcal{O}(\kappa \log(1/\epsilon))$ (width $\tilde{\Omega}(L r d_{out} \kappa^3)$)
[Zou et al., 2020]	Zero initialization (with residual connections)	$\mathcal{O}(\kappa \log(1/\epsilon))$ (width $\tilde{\Omega}(r d_{out} \kappa^2)$)

Main Takeaway

- Lower bounds can be overly pessimistic if the structure of the network is ignored.
- Depth may have no effect if the network is sufficiently wide or if skip connections are used.
- Initialization plays a crucial role in optimization.
- If a network is more expressive in terms of feature learning, a single GD step can reduce the loss significantly—blurring the lines between feature learning and optimization! :)

For Motivated Students

what happens if we use batch-normalization and residual connection ?

Thank You :)

References


Arora, S., Cohen, N., Golowich, N., and Hu, W. (2018).
A convergence analysis of gradient descent for deep linear neural networks.
arXiv preprint arXiv:1810.02281.


Bartlett, P., Helmbold, D., and Long, P. (2018).
Gradient descent with identity initialization efficiently learns positive definite linear transformations by deep residual networks.
In *International conference on machine learning*, pages 521–530. PMLR.


Du, S. and Hu, W. (2019).
Width provably matters in optimization for deep linear neural networks.
In *International Conference on Machine Learning*, pages 1655–1664. PMLR.


Shamir, O. (2019).
Exponential convergence time of gradient descent for one-dimensional deep linear neural networks.
In *Conference on Learning Theory*, pages 2691–2713. PMLR.


Zou, D., Long, P. M., and Gu, Q. (2020).
On the global convergence of training deep linear resnets.
arXiv preprint arXiv:2003.01094.