ON THE CONE EFFECT IN THE LEARNING DYNAMICS

Zhanpeng Zhou[1]*, Yongyi Yang[2], Jie Ren[1], Mahito Sugiyama[3,4], Junchi Yan[1]*
[1]Sch. of Computer Science & Sch. of Artificial Intelligence, Shanghai Jiao Tong University
[2]University of Michigan [3]National Institute of Informatics [4]The Graduate University for Advanced Studies, SOKENDAI
{zzp1012,yanjunchi}@sjtu.edu.cn

ABSTRACT

Understanding the learning dynamics of neural networks is a central topic in deep learning community. In this paper, we take an empirical perspective to study the learning dynamics of neural networks in practical, real-world settings. Our key findings reveal a two-phase learning process: i) Phase I, characterized by highly non-linear dynamics indicative of the rich regime, and ii) Phase II, where dynamics continue to evolve but are constrained within a narrow space, a phenomenon we term the cone effect. This two-phase framework builds on the hypothesis proposed by Fort et al. (2020), but we uniquely identify and analyze the cone effect in Phase II, demonstrating its significant performance advantages over fully linearized training.

1 INTRODUCTION

Research on the learning dynamics of neural network has drawn considerable attention in the learning theory community. For example, Jacot et al. (2018b); Chizat et al. (2019); Yang & Hu (2021); Arora et al. (2019b) study the infinitely wide neural networks, where the dynamics are linear and can be captured by a static kernel function, commonly referred to as the *lazy regime*. However, the lazy regime deviates significantly from the real-world scenarios and cannot account for the surprising generalization ability of neural networks (Chizat et al., 2019). A series of recent studies aim to uncover the complex non-linear learning dynamics beyond the lazy regime, namely the *rich regime* Geiger et al. (2020); Woodworth et al. (2020), but they generally focus on extremely simplified models. Studying the learning dynamics in practice remains an area of open research.

Our contributions. In this work, we investigate the learning dynamics of neural networks in real-world scenarios. Specifically, we found that:

- In **Phase I**, neural networks exhibit highly non-linear dynamics, signaling the rich regime.
- In **Phase II**, other than the lazy regime, the dynamics keep evolving but are constrained in a narrow space, namely the *cone effect*.

This two-phase picture are originally inspired from the integrated view of the learning dynamics from Fort et al. (2020), where they proposed the two-phase hypothesis (see details in Section 3). However, we surprisingly discover the cone effect in the second phase of the learning dynamics, and further show that this cone effect yields non-negligible performance benefits over completely linearized training.

Roadmap. This paper is organized as follows: Section 2 summarizes the related works on the learning dynamics, especially works on the lazy and rich regimes. Section 3 introduces the basic notations, the two-phase hypothesis in Fort et al. (2020), and main experimental setups. Section 4 delivers our main discovery on the cone effect.

2 RELATED WORK

Lazy Regimes. Numerous theoretical studies (Du et al., 2019b; Li & Liang, 2018; Du et al., 2019a; Allen-Zhu et al., 2019; Zou et al., 2020) proved that over-parameterized models can achieve zero training loss with minimal parameter variation. Moreover, Jacot et al. (2018a); Yang (2019); Arora

*Corresponding author.

et al. (2019b); Lee et al. (2019) showed that the learning dynamics of infinitely wide neural networks can be captured by a frozen kernel at initialization, known as the Neural Tangent Kernel (NTK). This behavior, often termed as *lazy regime*, typically occurs in over-parameterized models with large initialization and is considered undesirable in practice Chizat et al. (2019).

Rich Regimes. In contrast to the lazy regime, where learning dynamics are linear, the rich regime, or feature learning, exhibits complex nonlinear dynamics (Geiger et al., 2020; Jacot et al., 2021; Xu & Ziyin, 2024). Previous work has established that the transition between these regimes depends critically on initialization parameters. For instance, Geiger et al. (2020); Woodworth et al. (2020) showed that the absolute scale of network initializations governs this transition. Subsequent work further revealed that the relative scale of initializations (Azulay et al., 2021; Kunin et al., 2024) and their effective rank (Liu et al., 2024) can similarly induce feature learning. Beyond initializations, factors like weight decay (Lewkowycz & Gur-Ari, 2020; Jacot et al., 2022) and large learning rates (Lewkowycz et al., 2020; Ba et al., 2022) have also been shown to drive the rich regimes.

3 Background and Preliminaries

Notation and Setup. In this work, we focus on a classification task. Denote $[k] := \{1, 2, \cdots, k\}$. Let $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ be the training set of size n, where $\boldsymbol{x}_i \in \mathbb{R}^{d_0}$ represents the i-th input and $y_i \in [c]$ represents the corresponding target. Here, c is the number of classes. Let $f : \mathcal{D} \times \mathbb{R}^p \to \mathbb{R}$ be the neural network. Then $f(\boldsymbol{x}, \boldsymbol{\theta}) \in \mathbb{R}$ denotes the output of model f on the input $\boldsymbol{x}$, where $\boldsymbol{\theta} \in \mathbb{R}^p$ represents the model parameters. Let $\ell(f(\boldsymbol{x}_i, \boldsymbol{\theta}), y_i)$ be the loss at the i-th data point, simplified to $\ell_i(\boldsymbol{\theta})$. The total loss over the dataset $\mathcal{D}$ is then denoted as $\mathcal{L}_{\mathcal{D}}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^n \ell_i(\boldsymbol{\theta})$. Additionally, we use bold lowercase letters (e.g., $\boldsymbol{x}$) to denote vectors, and bold uppercase letters (e.g., $\boldsymbol{A}$) to represent matrices. For a matrix $\boldsymbol{A}$, let $\|\boldsymbol{A}\|_2$, $\|\boldsymbol{A}\|_F$, and $\mathrm{Tr}(A)$ denote its spectral norm, Frobenius norm, and trace, respectively.

Training Dynamics, Neural Tangent Kernel. Consider minimizing the total loss $\mathcal{L}_{\mathcal{D}}(\boldsymbol{\theta})$ using gradient flow; the update rule can be written as follows:

$$\frac{d\boldsymbol{\theta}(t)}{dt} = -\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}(t)) = -\sum_{i=1}^n \left(\frac{\partial f_t(\boldsymbol{x}_i)}{\partial \boldsymbol{\theta}}\right)^\top \frac{\partial \mathcal{L}_{\mathcal{D}}(\boldsymbol{\theta}(t))}{\partial f_t(\boldsymbol{x}_i)}, \tag{1}$$

where we denote $f_t(\boldsymbol{x}_i) := f(\boldsymbol{x}_i, \boldsymbol{\theta}(t))$ for simplicity. Subsequently, the evolution of the network output $f_t(\boldsymbol{x}_i)$ can be written as follows.

$$\frac{df_t(\boldsymbol{x}_i)}{dt} = -\sum_{j=1}^n \left\langle \frac{\partial f_t(\boldsymbol{x}_i)}{\partial \boldsymbol{\theta}}, \frac{\partial f_t(\boldsymbol{x}_j)}{\partial \boldsymbol{\theta}} \right\rangle \frac{\partial \mathcal{L}(\boldsymbol{\theta}(t))}{\partial f_t(\boldsymbol{x}_i)}. \tag{2}$$

Denoting $\boldsymbol{u}(t) = \{f_t(\boldsymbol{x}_i)\}_{i=1}^n \in \mathbb{R}^n$ as the network outputs for all inputs, then a more compact form of Equation (2) is given by:

$$\frac{d\boldsymbol{u}(t)}{dt} = -\boldsymbol{H}(\boldsymbol{\theta}(t))\nabla_{\boldsymbol{u}(t)}\mathcal{L}(\boldsymbol{\theta}(t)), \tag{3}$$

where $\boldsymbol{H}(\boldsymbol{\theta}(t)) \in \mathbb{R}^{n\times n}$ is a kernel matrix and its (i, j)-th entry is $\left\langle \frac{\partial f_t(\boldsymbol{x}_i)}{\partial \boldsymbol{\theta}}, \frac{\partial f_t(\boldsymbol{x}_j)}{\partial \boldsymbol{\theta}} \right\rangle$. This matrix is referred to as the *empirical neural tangent kernel (eNTK)*. For an infinitely wide neural network, the matrix $\boldsymbol{H}(\boldsymbol{\theta}(t)) \approx \boldsymbol{H}(\boldsymbol{\theta}(0))$ remains nearly constant throughout training (Arora et al., 2019a; Jacot et al., 2018b), which is also known as the **lazy regime** (Dominé et al., 2024).

In comparison, for neural networks with finite width, the eNTK matrix $\boldsymbol{H}(\boldsymbol{\theta}(t))$ evolves significantly during training, commonly termed as the **rich regime**. To quantify the changes in the eNTK matrix during training, Fort et al. (2020) introduced two metrics: *kernel distance* and *kernel velocity*. The kernel distance $S(\boldsymbol{\theta}, \boldsymbol{\theta}')$ measures the difference between the kernel matrices at two different points, i.e., $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$, in a scale-invariant way:

$$S(\boldsymbol{\theta}, \boldsymbol{\theta}') \triangleq 1 - \frac{\mathrm{Tr}(\boldsymbol{H}(\boldsymbol{\theta})\boldsymbol{H}^\top(\boldsymbol{\theta}'))}{\sqrt{\mathrm{Tr}(\boldsymbol{H}(\boldsymbol{\theta})\boldsymbol{H}^\top(\boldsymbol{\theta}))}\sqrt{\mathrm{Tr}(\boldsymbol{H}(\boldsymbol{\theta}')\boldsymbol{H}^\top(\boldsymbol{\theta}'))}} \tag{4}$$

Building on the kernel distance, the kernel velocity $v(t)$ captures the rate at which the kernel changes at a given iteration t:

$$v(t) \triangleq \frac{S(\boldsymbol{\theta}(t), \boldsymbol{\theta}(t + dt))}{dt} \tag{5}$$

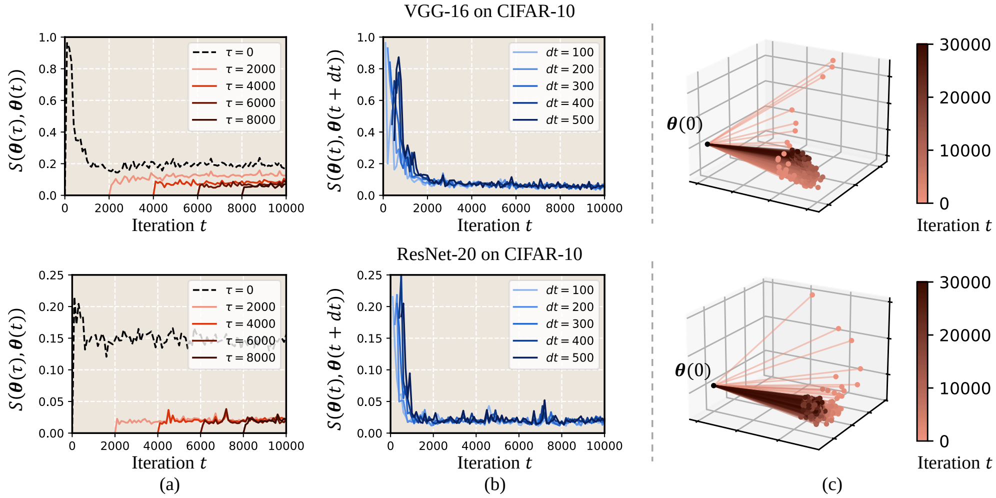

Figure 1: **The cone effect in the learning dynamics.** **(a)** The kernel distance between the current iterate $\boldsymbol{\theta}(t)$ and a reference point $\boldsymbol{\theta}(\tau)$ v.s. training iteration t, where τ is varied. **(b)** The kernel distance between two adjacent iterates $\boldsymbol{\theta}(t)$ and $\boldsymbol{\theta}(t+dt)$ vs. training iteration t, where dt is varied. **(c)** The visualization of the changes of the eNTK matrices $\boldsymbol{H}(t)$. The black dot represents position of the eNTK matrix at initialization, i.e., $\boldsymbol{H}(0)$. The other dot represents the relative position of $\boldsymbol{H}(t)$ at $t > 0$, with darker color indicating larger iteration.

A large kernel velocity indicates that the kernel is evolving rapidly during training, which is characteristic of the rich regime. Together, these metrics provide a framework for analyzing the dynamics of neural network training, offering implications for learning behavior.

The Two-Phase Hypothesis Fort et al. (2020) discovered a two-phase phenomenon in the training dynamics of neural networks via the *kernel velocity*. Specifically, they found that the kernel velocity is relatively large during the first few training epochs but drops rapidly after that. In addition, they showed that using standard training followed by linearized training performs similarly to using only standard training with a small learning rate. Based on these observations, they proposed the following two-phase hypothesis:

- *In **Phase I**, neural networks exhibit highly non-linear dynamics, signaling the rich regime.*
- *In **Phase II**, the dynamics are approximately linear, and the models enter the lazy regime.*

Main Experimental Setup. Following Frankle et al. (2020), we perform our experiments on commonly used image classification datasets MNIST (LeCun et al., 1998) and CIFAR-10 (Krizhevsky et al., 2009), and with the standard network architectures ResNet-20 (He et al., 2016), VGG-16 (Simonyan & Zisserman, 2015) and LeNet (LeCun et al., 1998). We follow the same training procedures and hyperparameters as in Frankle et al. (2020); Ainsworth et al. (2023).

4 The Cone Effect in Learning Dynamics

Previous studies empirically characterized the training dynamics of neural networks into two phases, transitioning from rich to lazy regimes. In this section, we dig deeper into this picture and observe that in the second phase, rather than the lazy regime, the training process evolves in a constrained function space, namely *the cone effect*.

Beyond the Lazy Regime: The Cone Effect. Through the kernel distance, we delve into the two phase hypothesis in the learning dynamics. First, we compute the distance between the kernel matrices at two adjacent points $\boldsymbol{\theta}(t)$ and $\boldsymbol{\theta}(t+dt)$. In Figure 1 *(b)*, we observe that, across different values of dt, the adjacent kernel distance $S(\boldsymbol{\theta}(t), \boldsymbol{\theta}(t+dt))$ is significant in the early training phase and then drops quickly to a low but non-negligible value. This result aligns with the two-phase hypothesis, where in Phase I (approximately $0 < t < 2000$) the model learns in the rich regime and kernel evolves significantly; in phase II, the model is in the lazy regime and kernel is approximately fixed. However, surprisingly, we note that in Phase II, the values of $S(\boldsymbol{\theta}(t), \boldsymbol{\theta}(t+dt))$ are upper-bounded

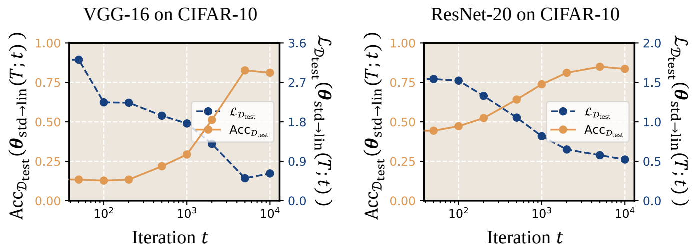

Figure 2: **The non-linear advantage of the cone effect.** Test accuracy $\text{Acc}_{\mathcal{D}_{\text{test}}}(\boldsymbol{\theta}_{\text{std}\to\text{lin}}(T;t))$ **(left)** and Test loss $\mathcal{L}_{\mathcal{D}_{\text{test}}}(\boldsymbol{\theta}_{\text{std}\to\text{lin}}(T;t))$ vs. the switching iteration t. $\boldsymbol{\theta}_{\text{std}\to\text{lin}}(T;t)$ represents the model initially trained with standard method up to iteration t, followed by linearized training up to iteration T, where T is set to 10^4.

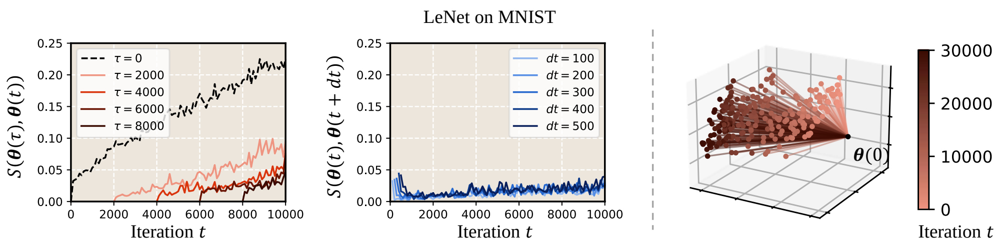

Figure 3: **A counterexample for the two-phase hypothesis.** **(a)** The kernel distance between the current iterate $\boldsymbol{\theta}(t)$ and a reference point $\boldsymbol{\theta}(\tau)$ v.s. training iteration t, where τ is varied. **(b)** The kernel distance between two adjacent iterates $\boldsymbol{\theta}(t)$ and $\boldsymbol{\theta}(t+dt)$ vs. training iteration t, where dt is varied. **(c)** The visualization of the changes of the eNTK matrices $\boldsymbol{H}(t)$. The black dot represents position of the eNTK matrix at initialization, i.e., $\boldsymbol{H}(0)$. The other dot represents the relative position of $\boldsymbol{H}(t)$ at $t > 0$, with darker color indicating larger iteration.

by a same value for different dt. One possible explanation to this phenomenon is that in Phase II, the kernel evolves in a constrained space. To validate this, we further measure how the distance between the kernel matrices at the current iterate $\boldsymbol{\theta}(t)$ and a referent point $\boldsymbol{\theta}(\tau)$ changes during training. As shown in Figure 1 *(a)*, for $\tau \in \{2000, 4000, 6000, 8000\}$, the kernel distance between the current iterate and the reference point, i.e., $S(\boldsymbol{\theta}(t), \boldsymbol{\theta}(\tau))$, first increases and then keeps nearly constant during training. This result verifies our claim and suggests that in Phase II, beyond the lazy regime, the model operates in a constrained function space. The visualization in Figure 1 *(c)* further confirms our picture, where a clear "cone" pattern is observed.

The Non-Linear Advantages of the Cone Effect. Despite the model evolving in a constrained function space in Phase II, it still provide significant advantages over completely lazy regime. To investigate this, we consider a "switching" training method: we first train a neural network with standardized training method, and switches to the linearized training (corresponds to the completely lazy regime) until T iterations. We vary the switching point t and obtain different solutions $\boldsymbol{\theta}_{\text{std}\to\text{lin}}(T;t)$. In Figure 2, we observe that the test performance of $\boldsymbol{\theta}_{\text{std}\to\text{lin}}(T;t)$ generally increases with t, especially when $t > 2000$. This result implies that the cone effect in Phase II still offers significant advantages over solely lazy regime.

Together, we conclude the following conjecture:

The Cone Effect: *In **Phase II**, beyond the lazy regime, the learning dynamics evolves in a constrained function space and performance improves significantly.*

5 Conclusion and Limitations

In summary, our work unveils a cone effect in the late learning dynamics. However, the cone effect and the two-phase hypothesis are not universal. In Figure 3, we observe a counterexample. Specifically, the kernel keeps evolving during training, and no clear two-phase pattern is observed. Further efforts on uncovering the contributing factors of the cone effect are expected.

References

Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha S. Srinivasa. Git re-basin: Merging models modulo permutation symmetries. In *ICLR*. OpenReview.net, 2023.

Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 242–252. PMLR, 09–15 Jun 2019.

Sanjeev Arora, Simon S. Du, Wei Hu, Zhiyuan Li, Ruslan Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. In *NeurIPS*, pp. 8139–8148, 2019a.

Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, Russ R Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019b.

Shahar Azulay, Edward Moroshko, Mor Shpigel Nacson, Blake E Woodworth, Nathan Srebro, Amir Globerson, and Daniel Soudry. On the implicit bias of initialization shape: Beyond infinitesimal mirror descent. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 468–477. PMLR, 18–24 Jul 2021.

Jimmy Ba, Murat A Erdogdu, Taiji Suzuki, Zhichao Wang, Denny Wu, and Greg Yang. High-dimensional asymptotics of feature learning: How one gradient step improves the representation. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022.

Lénaïc Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Clémentine C. J. Dominé, Nicolas Anguita, Alexandra M. Proca, Lukas Braun, Daniel Kunin, Pedro A. M. Mediano, and Andrew M. Saxe. From lazy to rich: Exact learning dynamics in deep linear networks. *CoRR*, abs/2409.14623, 2024.

Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1675–1685. PMLR, 09–15 Jun 2019a.

Simon S. Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. In *International Conference on Learning Representations*, 2019b.

Stanislav Fort, Gintare Karolina Dziugaite, Mansheej Paul, Sepideh Kharaghani, Daniel M. Roy, and Surya Ganguli. Deep learning versus kernel learning: an empirical study of loss landscape geometry and the time evolution of the neural tangent kernel. In *NeurIPS*, 2020.

Jonathan Frankle, Gintare Karolina Dziugaite, Daniel Roy, and Michael Carbin. Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning*, pp. 3259–3269. PMLR, 2020.

Mario Geiger, Stefano Spigler, Arthur Jacot, and Matthieu Wyart. Disentangling feature and lazy training in deep neural networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2020 (11):113301, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Arthur Jacot, Franck Gabriel, and Clement Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018a.

Arthur Jacot, Clément Hongler, and Franck Gabriel. Neural tangent kernel: Convergence and generalization in neural networks. In *NeurIPS*, pp. 8580–8589, 2018b.

Arthur Jacot, François Ged, Berfin Şimşek, Clément Hongler, and Franck Gabriel. Saddle-to-saddle dynamics in deep linear networks: Small initialization training, symmetry, and sparsity. *arXiv preprint arXiv:2106.15933*, 2021.

Arthur Jacot, Eugene Golikov, Clément Hongler, and Franck Gabriel. Feature learning in l_2-regularized DNNs: Attraction/repulsion and sparsity. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Daniel Kunin, Allan Raventos, Clémentine Carla Juliette Dominé, Feng Chen, David Klindt, Andrew M Saxe, and Surya Ganguli. Get rich quick: exact solutions reveal how unbalanced initializations promote rapid feature learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Jaehoon Lee, Lechao Xiao, Samuel Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Aitor Lewkowycz and Guy Gur-Ari. On the training dynamics of deep networks with l_2 regularization. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 4790–4799. Curran Associates, Inc., 2020.

Aitor Lewkowycz, Yasaman Bahri, Ethan Dyer, Jascha Sohl-Dickstein, and Guy Gur-Ari. The large learning rate phase of deep learning: the catapult mechanism. *arXiv preprint arXiv:2003.02218*, 2020.

Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Yuhan Helena Liu, Aristide Baratin, Jonathan Cornford, Stefan Mihalas, Eric Todd SheaBrown, and Guillaume Lajoie. How connectivity structure shapes rich and lazy learning in neural circuits. In *The Twelfth International Conference on Learning Representations*, 2024.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1409.1556.

Blake Woodworth, Suriya Gunasekar, Jason D. Lee, Edward Moroshko, Pedro Savarese, Itay Golan, Daniel Soudry, and Nathan Srebro. Kernel and rich regimes in overparametrized models. In Jacob Abernethy and Shivani Agarwal (eds.), *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pp. 3635–3673. PMLR, 09–12 Jul 2020.

Yizhou Xu and Liu Ziyin. When does feature learning happen? perspective from an analytically solvable model. *arXiv preprint arXiv:2401.07085*, 2024.

Greg Yang. Scaling limits of wide neural networks with weight sharing: Gaussian process behavior, gradient independence, and neural tangent kernel derivation. *arXiv preprint arXiv:1902.04760*, 2019.

Greg Yang and Edward J. Hu. Tensor programs iv: Feature learning in infinite-width neural networks. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 11727–11737. PMLR, 18–24 Jul 2021.

Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Gradient descent optimizes over-parameterized deep relu networks. *Machine learning*, 109:467–492, 2020.