Toward Disentangling Double Descent and Information Flow in Deep Neural Networks

Chris Liu Brendan King Jing Gu

November 29, 2022

University of California, Santa Cruz

Table of Contents

1. Introduction

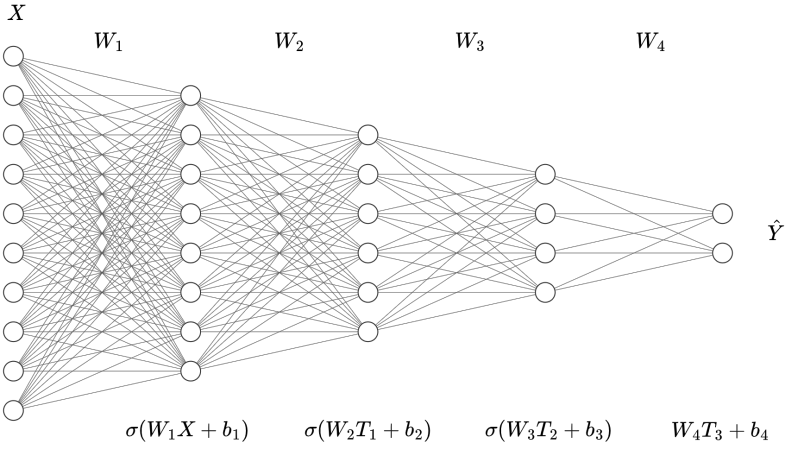
2. Background

3. Experiments

4. Results

5. Summary

Neural Networks



Neural Networks (Cont.)

An m-layer neural network f is a composition of linear functions $\{l_0, l_1, ...l_{m-1}\}$ and non-linear functions $\{\sigma_0, \sigma_1, ..., \sigma_{m-1}\}$[1] with parameters (weights) $\{\theta_0, \theta_1, ..., \theta_{m-1}\}$. More concisely,

$$f = \sigma_{m-1} \circ l_{m-1} \circ \cdots \circ \sigma_2 \circ l_2 \circ \sigma_1 \circ l_1,$$

where f has the form $f(\cdot; \theta)$ and θ is distributed in $\{l_0, l_1, ...l_{m-1}\}$.

[1] The last nonlinearity is usually omitted.

Notation

- θ: parameters (weights) of a function (model)
- $f(\cdot;\theta)$: a function parameterized by θ
- $\mathcal{D} = \{(x_k, y_k)\}_{k=1}^{N} \overset{\text{i.i.d.}}{\sim} p(X, Y)$: a collection of N i.i.d. data points
- ϵ: error function
- $\mathcal{L}$: loss function
- L: aggregated loss

$$\text{Error rate} \quad \epsilon\left(f_\theta, \mathcal{D}\right) = \mathbb{E}_{(x,y)\sim p(X,Y)}\, \mathbb{1}[f(x;\theta) \neq y] \approx \frac{1}{N}\sum_{i}^{N} \mathbb{1}\left[f\left(x_i;\theta\right) \neq y_i\right]$$

$$\text{Loss} \quad L\left(f_\theta, \mathcal{D}\right) = \mathbb{E}_{(x,y)\sim p(X,Y)}\, L(f(x;\theta), y) \approx \frac{1}{N}\sum_{i}^{N} \mathcal{L}\left(f\left(x_i;\theta\right), y_i\right)$$

Table of Contents

1. Introduction

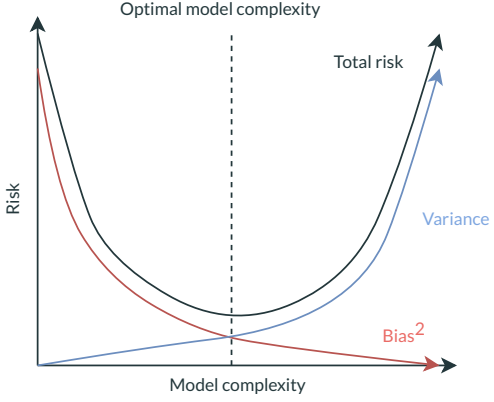
2. Background

3. Experiments

4. Results

5. Summary

Bias-Variance Trade-off



$$\text{Expected total risk} = \text{variance} + \text{bias}^2 + \text{irreducible error}$$

Such a decomposition is not always available for loss functions other than MSE, but the estimated quantites still follow the equation above.

The Central Question in Deep Learning

Why do overparameterized deep neural networks generalize well in practice?

Previous Attempts to Answer the Question

1. **Implicit (regularization) bias in SGD** (Zhang et al., 2016; Soudry et al., 2018; Nacson et al., 2019): SGD optimizes toward a max-margin solution
2. **Neural tangent kernel** (Jacot et al., 2018): infinite-width neural networks accuractly approximates their (linear) first-order Taylor expansion
3. **Lottery ticket hypothesis** (Frankle and Carbin, 2019): existence of (tiny) sparse sub-networks that outperform the whole network
4. ...

Definition of Double Descent

An ordered set of parameters

Given a set of functions $f^M = \{f_0, f_1, ..., f_M\}$ with their corresponding parameters $\theta^M = \{\theta_0, \theta_1, ..., \theta_M\}$ indexed by i. For any two indices $i, j \in \mathbb{Z}^+$ and $0 \leq i \leq M$ and $0 \leq j \leq M$, if $i < j$, we must have $|\theta_i| < |\theta_j|$. The converse also hold.

For notation convenience, we also let $\mathcal{I} : \mathbb{Z}^+ \to \mathbb{Z}^+$ be an indexing function that takes as input $|\theta|$ (where $\theta \in \theta^M$) and returns its index i.

Definition of Double Descent (Cont.)

Interpolation threshold

The interpolation threshold (IT) is located at $|\theta| = N$ (N is the dataset size) with a certain degree of drifting (α, β) such that $|\theta| - \alpha \leq \mathrm{IT} \leq |\theta| + \beta$[1].

Optimal index

The optimal index is the index of a function $f(\cdot; \theta)$ (or parameters θ) such that $\min_{0 \leq i \leq \mathcal{I}(\mathrm{IT})} \epsilon\left(f_{\theta_i}, \mathcal{D}\right)$ or $\min_{0 \leq i \leq \mathcal{I}(\mathrm{IT})} L\left(f_{\theta_i}, \mathcal{D}\right)$ is obtained.

[1] This is also known as the critical regime (Nakkiran et al., 2020).

Definition of Double Descent (Cont.)

Double descent

Double descent occurs when the expected risk (i.e., ϵ or L) follows a U-shaped or monotone non-decreasing curve for underparameterized models and decreases monotonically for overparameterized models. A double descent curve is *singly-peaked* if the optimal index is equal to 0, and is *doubly-peaked* if the optimal index satisfies $0 < i < \mathcal{I}(\mathrm{IT})$.

Singly- vs. Doubly-peaked double descents

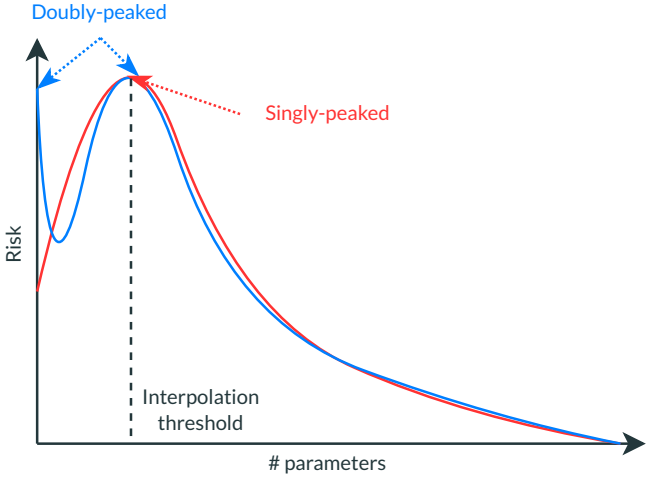


Figure 1: The singly- and doubly-peaked double descent curves were later identified in Yang et al. (2020) due to different dominating stages of bias and variance.

A Real Example of Double Descent

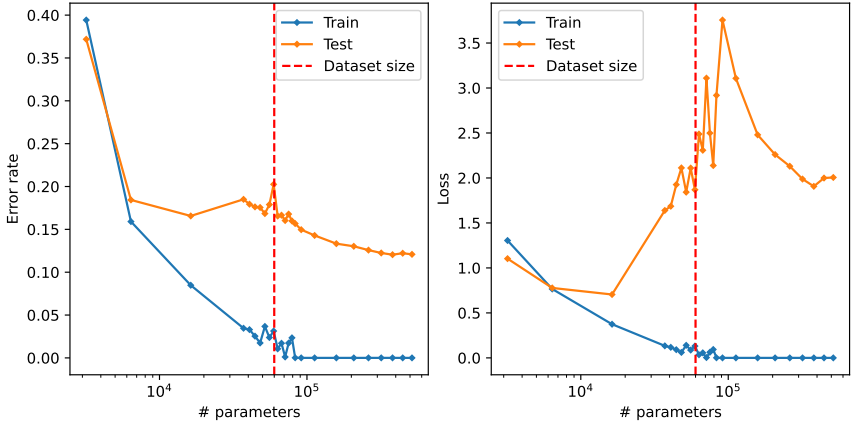
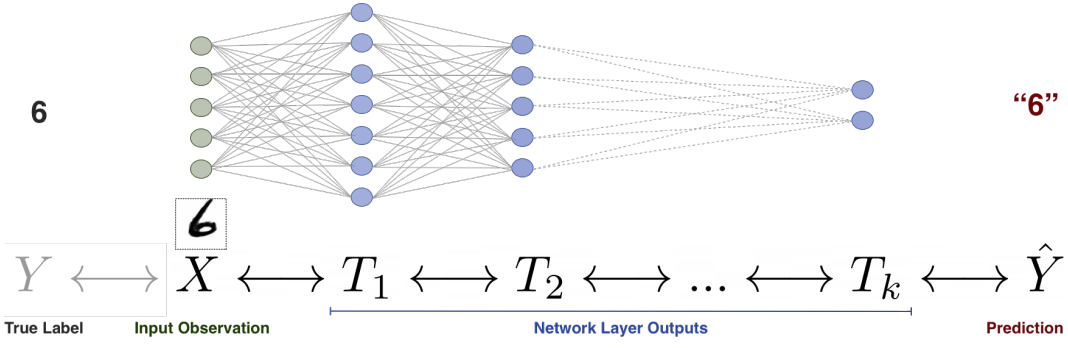


Figure 2: When the number of parameters of a network is smaller than a certain threshold, the expected risk follows the traditional U-shaped curve; if the size of a model continues growing, the expected risk experiences a second descent. The plot is generated by a fully-connected network with ELU on Fashion-MNIST.

Neural Networks as Markov Chains



Data Processing Inequality

Following the previous Markov chain formulation $Y \leftrightarrow X \leftrightarrow T$, we have the data processing inequality below:

$$I(X;Y) \geq I(T_1;Y) \geq I(T_2;Y) \geq \ldots \geq I(T_k;Y) \geq I(\hat{Y};Y)$$
$$H(X) \geq I(X;T_1) \geq I(X;T_2) \geq \ldots \geq I(X;T_k) \geq I(X;\hat{Y}).$$

The above holds with equality when T_i is a sufficient statistics of X or its previous representation T_{t-1}. Therefore, $I(Y;T_i)$ needs to be maximized and $I(T_i;T_{i-1})$ needs to be minimized.

Information Bottleneck Principle

A sufficient statistics of X w.r.t. Y is

$$I(S(X); Y) = I(X; Y).$$

A minimal sufficient statistics $T(X)$ can then be written as

$$T(X) = \arg \min_{S(X): I(S(X);Y)=I(X;Y)} I(S(X); X).$$

With Lagrangian relaxation,

$$\min_{p(t|x),p(y|t),p(t)} \{I(X; T) - \beta I(T; Y)\}.$$

We omit the irrelevant detail here, but the optimization problem can be cast as a constrained maximization problem as follows:

$$\max_{\theta} I(T; Y)$$
$$\text{s.t. } I(X; T) \leq I_c$$

The Information Plane

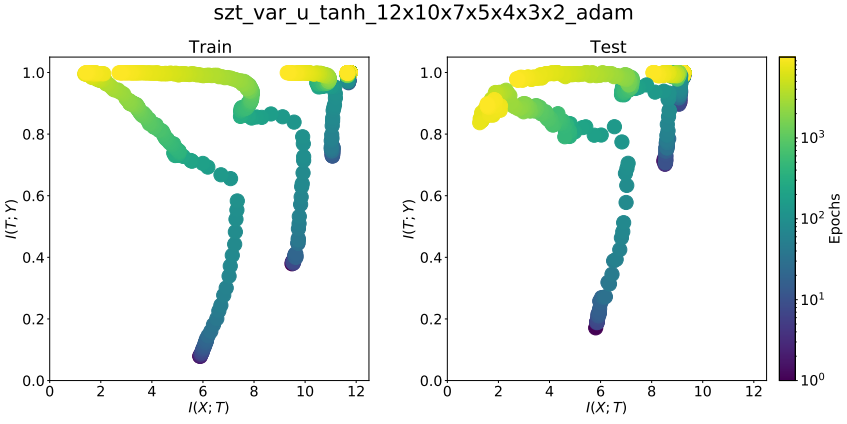


Figure 3: Phase 1: Both $I(X;T)$ and $I(T;Y)$ increase. Phase 2: $I(T;Y)$ increases, but $I(X;T)$ decreases.

Challenge: Estimating Mutual Information

- We aim to estimate the mutual information between each of $\{T_1, T_2, ...T_k\}$ and both X and Y.
- **Problem**: estimation is inherently difficult [1]
  - $\mathcal{X}, \mathcal{T}_i$ are continuous, high-dimensional spaces
  - We only have N samples with which to estimate probability densities $f(\cdot)$

below

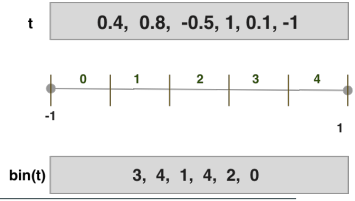
$$I(X; T_i) = \int f(x, t_i) \, log \left(\frac{f(x, t_i)}{f(x) f(t_i)} \right) dx dt_i$$

[1] This is discussed at length in Paninski (2003).

Binning Method[1]

Solution: utilize the data processing inequality. Compute a lower bound $\hat{I}(X; \text{Bin}(T_i)) \leq I(X; T_i)$ using a discrete approximation of t_i given by a binning function Bin:

$$\hat{I}(X; T_i) = \sum_{\mathcal{X} \times \text{Bin}(\mathcal{T}_i)} \hat{P}(x, \text{Bin}(t_i)) \; log \left(\frac{\hat{P}(x, \text{Bin}(t_i))}{\hat{P}(x) \times \hat{P}(\text{Bin}(t_i))} \right)$$



$$\hat{P}(\text{Bin}(t_i)) = \frac{count(\text{Bin}(t_i) = ...)}{N}$$

$$\hat{P}(x) = \frac{count(x = ...)}{N}$$

[1] We use this method as it is simple to implement and used in Shwartz-Ziv and Tishby (2017). Other methods exist, but analyses using different estimators cannot be directly compared (Geiger (2020)).

Research Questions

1. Is the amount of compression an indicator of good generalization? If not, what is?
2. Do neural networks in different sizes behave the same during training in terms of compression?
3. What, in terms of mutual information compression (phase transition), distinguishes the two stages on a double descent curve?

Table of Contents

1. Introduction

2. Background

3. Experiments

4. Results

5. Summary

Experimental Setup

Random (non-linear) feature networks

$$\mathbf{Z}_i = \max\left(\langle \mathbf{w}_i, \mathbf{X} \rangle, 0\right), \quad i = 1, ..., M,$$

$$\mathbf{w}_i \sim \mathcal{N}(0, \sigma^2), \quad \sigma = \sqrt{\frac{1}{\mathrm{Dim}(X)}}$$

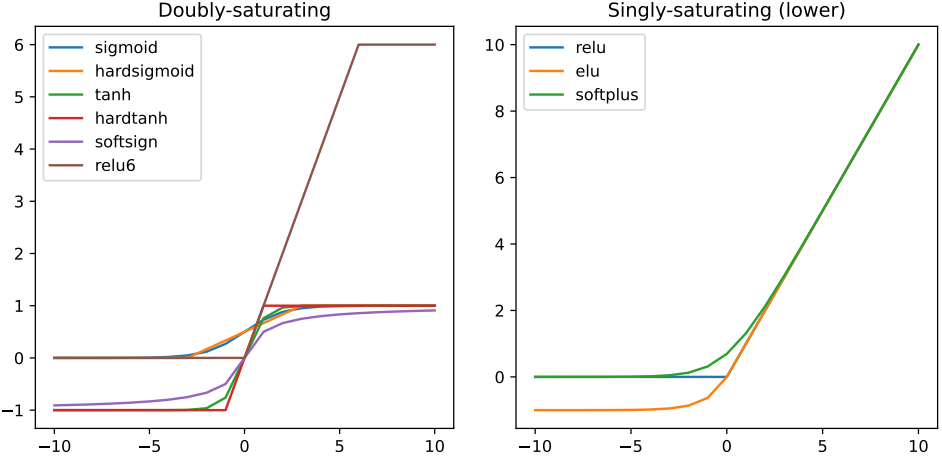
Fully-connected networks with 5 layers and base width W:

$$(\mathrm{Dim}(X), 8W, 4W, 2W, W, C)$$

$$\mathbf{w}_i \sim \mathcal{U}\left(-\sqrt{\frac{1}{\mathrm{Dim}(X)}}, \sqrt{\frac{1}{\mathrm{Dim}(X)}}\right)$$

Experimental Setup (Cont.)

Activation functions (for both non-linear features and fully-connected networks)



Experimental Setup (Cont.)

- All eperiments employ random feature networks and fully-connected networks using SGD with constant learning rate on MNIST, Fashion-MNIST, and CIFAR-10.
- When producing double descent, we only vary the base width of the network and train them with the setting above.
- The bin size used for mutual information estimation is always 30, following Shwartz-Ziv and Tishby (2017) and Saxe et al. (2018).
- The range of the bins is determined by the particular activation function (i.e., [-1, 1] for hyperbolic tangent, [0, max) for ReLU).

Table of Contents

1. Introduction

2. Background

3. Experiments

4. Results

5. Summary

Key Results

1. A higher compression of $I(X;T)$ in some linear models corresponds perfectly to good generalization, but it is the opposite for (large) neural networks.
2. The good generalization of neural networks cannot be directly explained by *how much* the compression of $I(X;T)$ is, but it can be related to *when* the compression happens.
3. The behavior of the phase transition of the last layer in an overparameterized neural network resembles that of a linear model, which is reflected by the signal-to-noise ratio of the gradients during training.

Compression in Linear Networks

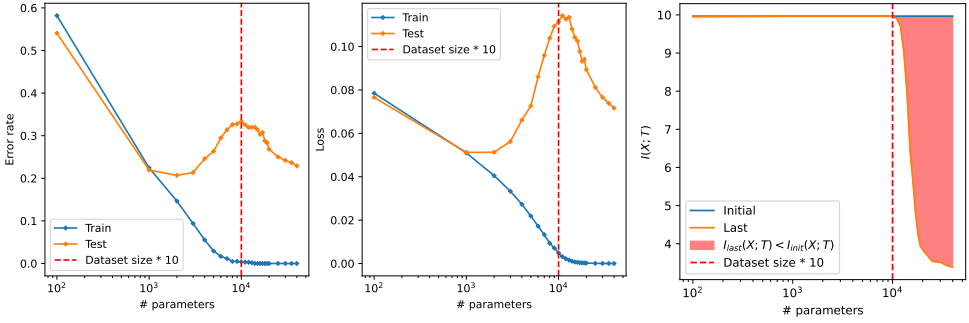
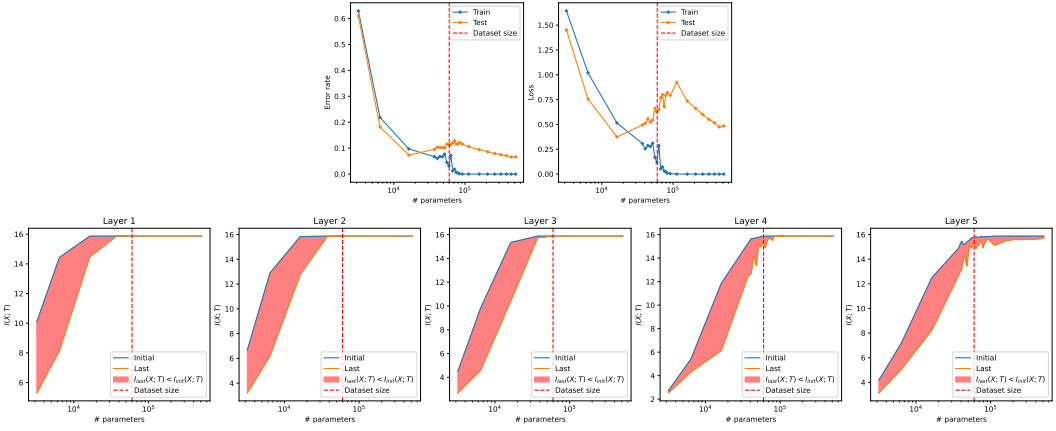


Figure 4: Only overparameterized linear networks compress, and the amount of compression is positively related to the number of parameters in the network.

Compression in Fully-Connected Networks



Drift Phase and Diffusion Phase

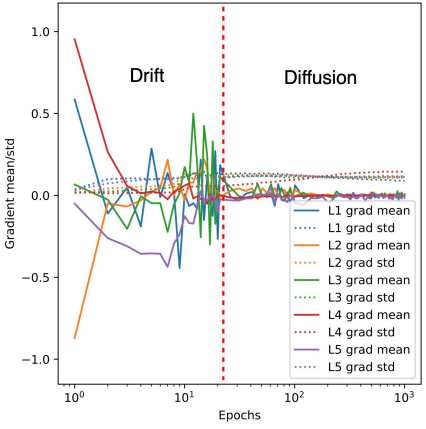


Figure 5: Proposed by Shwartz-Ziv and Tishby (2017), in the *drift phase*, the gradient means are larger than their standard deviations, indicating stable gradient flow and low stochasticity. In the diffusion phase, the gradient standard deviations are larger than the means. These two phases correspond to high and low signal-to-noise ratio, respectively.

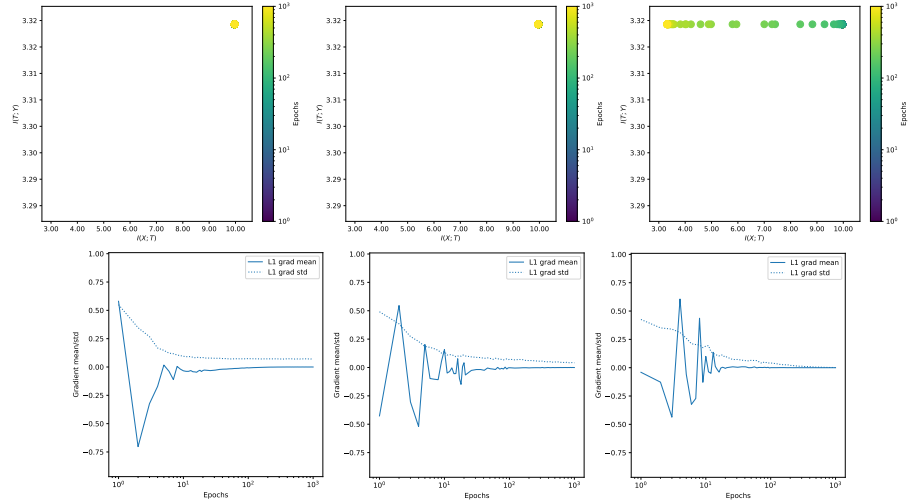
Drift Phase and Diffusion Phase (Cont.)

In Shwartz-Ziv and Tishby (2017), the authors prove that the stationary distribution of a diffusion process governed by a Focker-Planck equation maximizes the conditional entropy $H(X \mid T_i)$ or minimizes the mutual information

$$I(X; T_i) = H(X) - H(X \mid T_i).$$

Note that $H(X)$ is fixed because the dataset does not change during training. The entropy maximization of $H(X \mid T_i)$ is also known as *a stochastic relaxation with low training error constraint*. Therefore, the diffusion phase leads to more compression in $I(X; T_i)$.

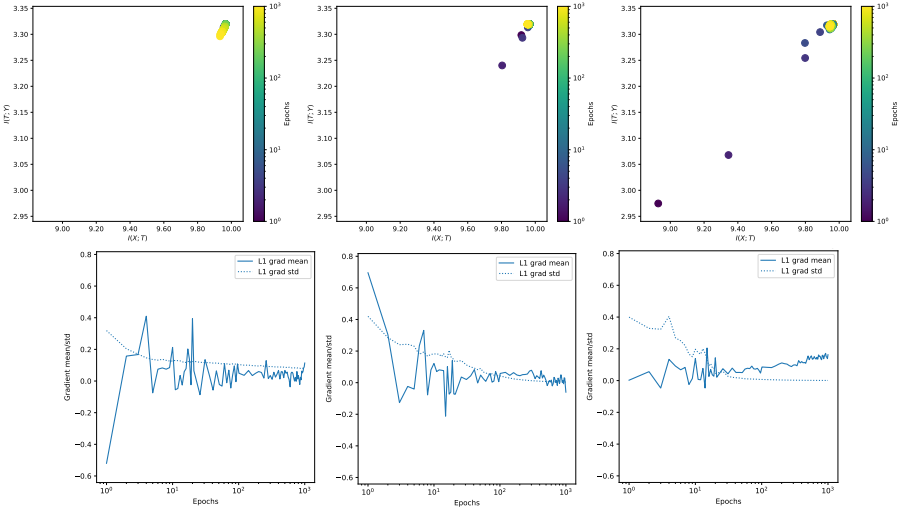
Phase Transition in Linear Networks with MSE Loss



(a) Underparameterized **(b)** $|\theta| \approx N$ **(c)** Overparameterized

Figure 6: Linear networks with MSE loss almost always stays in the diffusion phase.

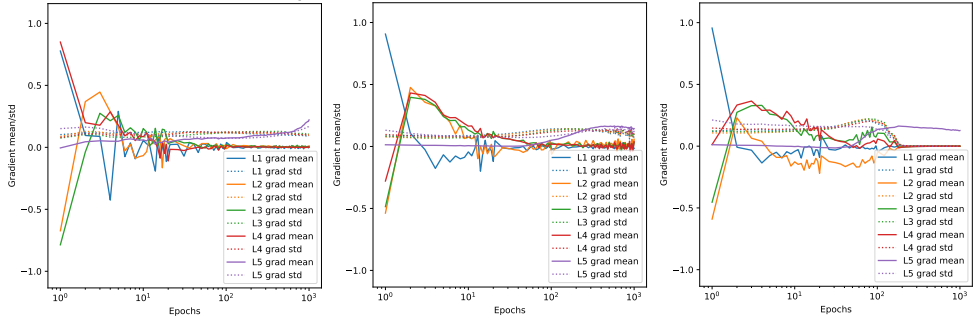
Phase Transition in Linear Networks with CE Loss



(a) Underparameterized **(b)** $|\theta| \approx N$ **(c)** Overparameterized

Figure 7: Large linear networks with CE loss go back to drift phase.

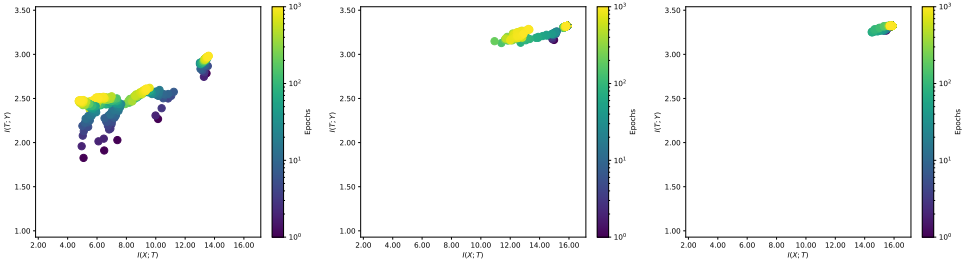
Phase Transition in Fully-Connected Networks



(a) Underparameterized **(b)** $|\theta| \approx N$ **(c)** Overparameterized

Figure 8: The left, middle, and right figures correspond to neural networks with size smaller, equal, and larger than the dataset size, respectively. The small network undergoes drift and diffusion phases; the medium network stays in the diffusion phase but has noisy gradient mean and standard deviation; the large network escapes the diffusion phase. Networks in (b) and (c) have the same training error (i.e., 0).

Phase Transition in Fully-Connected Networks (Cont.)



(a) Underparameterized **(b)** $|\theta| \approx N$ **(c)** Overparameterized

Figure 9: Left: All layers still compress. Middle: $I(X;T)$ in the last layer compresses, but other layers do not move. Right: $I(X;T)$ in the last layer expands, all other layers becomes almost stationary.

Table of Contents

1. Introduction

2. Background

3. Experiments

4. Results

5. Summary

Phase Transition Summary

For linear networks:

- Linear models with MSE loss enter the drift phase and stay in the diffusion phase, resulting in low SNR and more compression;
- Linear models with cross-entropy loss always stay in the drift phase, maintaining high SNR and low (or no) compression.

Phase Transition Summary

For fully-connected networks:

- Underparametrized networks enter the diffusion phase and stays $\rightarrow$ more compression;
- $|\theta| \approx N$ networks behaves similar to underparameterized ones, but they generally compress less and have noisy gradients;
- Overparameterized networks experience the diffusion phase for a short amount of time and immediately becomes stationary except for the last layer, which possibly yields high SNR.
- Singly-saturating nonlinearities are prone to expansion in $I(X;T)$, whereas doubly-saturating activations still preserve slight compression for large network sizes.
- The "stationary" phase is unique to overparameterized networks.

Connection with Double Descent and Generalization

- The amount of compression in $I(X;T)$ is NOT a good proxy for generalization.
- The point when phase transition occurs and the noisiness of the gradient distinguishes underparameterized, $|\theta| \approx N$, and overparameterized networks.
  - Underparameterized: all layers compress
  - $|\theta| \approx N$: only the last layer compresses
  - Overparameterized: the last layer expands, all other layers become stationary

Unanswered Questions

1. What makes large neural networks revert to the drift phase (for the last layer)? Why is this behavior unique to overparameterized networks?
2. Is such a phase reversion intrinsic to large neural networks? Can we enforce it?
3. Why, even without the presence of explicitly regularization, do overparameterized networks generalize well?

References i

Frankle, J. and Carbin, M. (2019). The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Geiger, B. C. (2020). On Information Plane Analyses of Neural Network Classifiers – A Review. *ArXiv preprint*, abs/2003.09671.

Jacot, A., Hongler, C., and Gabriel, F. (2018). Neural tangent kernel: Convergence and generalization in neural networks. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 8580–8589.

Nacson, M. S., Srebro, N., and Soudry, D. (2019). Stochastic gradient descent on separable data: Exact convergence with a fixed learning rate. In Chaudhuri, K. and Sugiyama, M., editors, *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pages 3051–3059. PMLR.

Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., and Sutskever, I. (2020). Deep double descent: Where bigger models and more data hurt. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Paninski, L. (2003). Estimation of entropy and mutual information. *Neural computation*, 15(6):1191–1253.

References iii

Saxe, A. M., Bansal, Y., Dapello, J., Advani, M., Kolchinsky, A., Tracey, B. D., and Cox, D. D. (2018). On the information bottleneck theory of deep learning. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Shwartz-Ziv, R. and Tishby, N. (2017). Opening the black box of deep neural networks via information. *ArXiv preprint*, abs/1703.00810.

Soudry, D., Hoffer, E., Nacson, M. S., and Srebro, N. (2018). The implicit bias of gradient descent on separable data. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Yang, Z., Yu, Y., You, C., Steinhardt, J., and Ma, Y. (2020). Rethinking bias-variance trade-off for generalization of neural networks. In *International Conference on Machine Learning*, pages 10767–10777. PMLR.

References iv

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2016). Understanding deep learning requires rethinking generalization. corr abs/1611.03530. *ArXiv preprint*, abs/1611.03530.