Dynamics of Gradient Flow with Contrastive Learning

by

Cem Tepe

B.S. Electrical Engineering and Computer Science, MIT, 2023

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2024

Authored by: Cem Tepe
Department of Electrical Engineering and Computer Science
May 10, 2024

Certified by: Navid Azizan
Assistant Professor, Thesis Supervisor

Accepted by: Katrina LaCurts
Chair
Master of Engineering Thesis Committee

Dynamics of Gradient Flow with Contrastive Learning

by

Cem Tepe

Submitted to the Department of Electrical Engineering and Computer Science
on May 10, 2024 in partial fulfillment of the requirements for the degree of

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

ABSTRACT

Contrastive learning (CL), in different forms, has been shown to learn discriminatory representations for downstream tasks without the need of human labeling. In the representation space learnt via CL, each class collapses to a distinct vertex of a simplex on a hypersphere during training. This property, also seen in other types of learning tasks, might explain why CL works as well as it does. Having class collapse on the test distribution, which determines how well the model generalizes to new samples and new classes, is tied to class collapse on the training distribution under certain conditions as studied by Galanti et al. (2022). In the case of CL, minimizing the contrastive loss has been shown to lead to collapse during training by Graf et al. (2021). In a recent study, Xue et al. (2023) show that the minimizing the contrastive loss is not enough to observe class collapse in the representation space for a single layer linear model and that we need minimum norm minimizers for the collapse to happen. However, their results don't explain how class collapse can occur without adding an explicit bias. The implicit bias of the gradient descent is a likely candidate to explain this phenomena. Here, we investigate the gradient flow of the spectral contrastive loss and give a theoretical description of the learning dynamics.

Thesis supervisor: Navid Azizan
Title: Assistant Professor

Acknowledgments

I would like to thank my MEng thesis supervisor Professor Navid Azizan, for his support and insightful discussions. His guidance allowed me to keep on track and provided me with new perspectives when I was stuck. I would like to thank my parents, Cavidan and Ali, for their emotional support, and my mentor V.R. Dimitrov for his hard work in teaching me and giving me the tools to succeed.

Contents

List of Figures

Chapter 1

Introduction

Contrastive learning (CL) was proposed as a form of self-supervised learning (SSL) where augmentations are used to artificially create similarities between different samples without needing any human labeled examples. For each sample from the original data, a set of randomly selected augmentations are applied to create a new dataset in which a pair is said to be similar if they were augmented from the same original sample. Using this dataset, a representation is learned by trying to pull together the similar points while trying to separate the rest of the augmented data. In the last few years, CL and its variations have been the most common framework for SSL for computer vision and natural language processing due to their performance and simplicity. Modern versions of self-supervised CL are usually derived from the SimCLR framework introduced by Chen et al. (2020).

Although initially proposed for SSL, CL is extended to the supervised setting and is shown to improve generalizability in certain situations (Khosla et al., 2020; Graf et al., 2021). Compared to the cross-entropy loss, supervised CL is shown to be more robust to data corruptions and less sensitive to hyperparameter selection. Although these properties of CL are demonstrated experimentally in both supervised and self-supervised settings, it is still not clear why it works as well as it does.

One domain where CL is proven to be useful is out-of-distribution (OOD) detection,

particularly for detecting new classes that were not seen during training. Tack et al. (2020) determine that for a model trained with CL, augmentation functions used to generate the data have a large impact on the detection performance of the model. However, the properties of models trained with CL when applied to the test data is still not properly understood, which is necessary to analyze their OOD detection performance theoretically.

The form of the CL loss functions allows us to make a connection between CL and the matrix factorization problem. At a high-level, this is due to the fact that the goal of CL is to learn correlations in the latent space in the absence of labels. Therefore CL tries to learn an unknown matrix that describes these correlations, usually given by some similarity metric, given the augmented data covariance. This can be made more concrete by analyzing the loss function for the matrix factorization problem and reducing the loss function of CL to a similar form. Using this connection, we can use the previous works on matrix factorization to understand the implicit bias of the gradient flow on CL loss function. In particular, we show that this bias can be used to explain a finding from Tack et al. (2020), who show that the l_2-norm of the representations can be used to determine if a sample is in or out-of-distribution at test time.

1.1 Contrastive Loss Functions

1.1.1 Self-supervised InfoNCE

The most common framework for self-supervised CL is called SimCLR, proposed by Chen et al. (2020), which uses negative cross-entropy (InfoNCE) type of loss to maximize the likelihood of positive samples. In this setting, two augmentations are generated for each original sample, and one of the augmentations is used as an anchor to define the InfoNCE loss.

$$\mathcal{L}_{CE} = -\frac{1}{N^2} \sum_{x,x^+} \log \frac{\exp\left(\langle f(x), f(x^+)\rangle / \tau\right)}{\sum\limits_{x^- \in \mathcal{D}_{aug} \setminus \{x^+\}} \exp\left(\langle f(x), f(x^-)\rangle / \tau\right)} \tag{1.1}$$

Here N is the size of the original dataset, $|\mathcal{D}_{aug}| = 2N$ is the set of all data augmentations generated, and the outer sum is over all positive pairs in $\mathcal{D}_{aug}$. For the InfoNCE loss, the mappings $f(x)$ are usually assumed to be normalized, i.e. $\|f(x)\|_2 = 1$.

1.1.2 Supervised InfoNCE

Let $X = (x_1, \ldots, x_N) \in \mathcal{X}^N$ be the data, where x are drawn from a distribution $\mathcal{P}$ given by their classes $y(x) \in [K]$ that are assumed to be known. Let $f : \mathcal{X} \to \mathcal{Z}$ be the encoder and $\mathcal{Z} \subset \mathbb{R}^d$ be the representation space.

$$\mathcal{L}_{SupCon} = -\sum_i \frac{1}{|C(i)|} \sum_{j \in C(i)} \log \frac{\exp\left(\langle f(x_i), f(x_j)\rangle / \tau\right)}{\sum_{k \neq i} \exp\left(\langle f(x_i), f(x_k)\rangle / \tau\right)} \tag{1.2}$$

Where i denotes a single sample and $C(i)$ denotes the set of samples that have the same class as i. Khosla et al. (2020) show that their proposed SupCon loss outperforms cross-entropy loss applied to the space of the probabilities in most of the tasks that they include in testing. They also show that SupCon improves the models robustness to data corruptions and its sensitivity to hyperparameters.

1.1.3 Spectral Contrastive Loss

Since contrastive learning depends only on the similarities between representations, it can be understood as learning a graph on the representation space. The graph edges give the similarity between two representations, and the goal is to only maximize edge weights that correspond to similar pairs. Then, ideally, each class would correspond to a connected component of the graph, or in the self-supervised setting, each learned connected component would denote a unique class. This property of CL allows HaoChen et al. (2021) to define the Spectral Contrastive Loss (SCL) in the following way.

$$\mathcal{L}_{SL} = -2\hat{\mathbb{E}}_{x,x^+}\left[f(x)^T f(x^+)\right] + \hat{\mathbb{E}}_{x,x' \sim \mathcal{D}_{aug}}\left[\left(f(x)^T f(x')\right)^2\right] \tag{1.3}$$

where $\hat{\mathbb{E}}$ denotes the sample mean. In the self-supervised setting, x, x^+ pairs are the augmented pairs from the same original sample, and x, x^- are any pair from the set of augmentations $\mathcal{D}_{aug}$. Although this loss is similar to InfoNCE for the positive pairs, the negative pairs in SPL are given by $\hat{\mathbb{E}}_{x,x' \sim \mathcal{D}_{aug}} \left[\left(f(x)^T f(x') \right)^2 \right]$ whereas in InfoNCE, the negative terms in the loss are $\hat{\mathbb{E}}_{x \sim \mathcal{D}_{aug}} \left[\log \hat{\mathbb{E}}_{x' \sim \mathcal{D}_{aug}} \left[e^{\left(f(x)^T f(x') \right)} \right] \right]$.

1.2 Matrix Factorization

Following the notation in Gunasekar et al. (2017), we can write the general matrix factorization problem as

$$\min_{X \succeq 0} F(X) := \|\mathcal{A}(X) - y\|_2^2 \tag{1.4}$$

for $X \in \mathbb{R}^{n \times n}$, $\mathcal{A} : \mathbb{R}^{n \times n} \to \mathbb{R}^m$ linear, and $y \in \mathbb{R}^m$. In the case of matrix sensing, $\mathcal{A}(X)_i = \text{tr}(A_i^T X) := \langle A_i, X \rangle$, where A_i are called detection matrices. Then, in the symmetric factorization case, the loss is in the form of

$$\mathcal{L}(W) = F(W^T W) = \sum_i (\langle A_i, W^T W \rangle - y_i)^2 \tag{1.5}$$

Another loss function that will be relevant to the upcoming analysis is related to the matrix completion problem. Here, instead of detection matrices, we are given an incomplete ground truth U^*, which corresponds to having $A_i = e_k e_l^T$ where e_k are the canonical basis unit vectors. This loss is given by,

$$\mathcal{L}(W) = F(W^T W) = \|W^T W - U^*\|_F^2 \tag{1.6}$$

1.3 Implicit Bias

When optimizing a loss function with an over-parameterized model, the space of minimizers is non-trivial, leading to an infinite number of solutions. To avoid such drawbacks of over-

parameterization, a regularization term is usually added to the loss function to create a bias towards certain solutions within this space. However, it has been shown that in some cases, the optimization algorithm itself creates a bias without any explicit regularization terms, which is referred to as the implicit bias of the algorithm. In particular, the implicit bias of the gradient descent algorithm (GD) is well-studied for various types of regression and classification tasks, with losses like mean-squared error (MSE) and cross-entropy. In the case of matrix factorization, there has been good progress in the last few years towards describing the implicit bias of GD. Despite the progress, the results are still not as definitive as the previously mentioned losses. It is generally argued that the reason for this is because there is no simple description for what is implicitly minimized by GD for matrix factorization, and instead, it can only be described in terms of the properties of its learning dynamics. In a similar sense, we will try to describe the learning dynamics of gradient flow for the Spectral Contrastive Loss.

Chapter 2

Background

Here, we first write down our assumptions about the data and introduce some notation that will be necessary for analyzing the gradient flow equation of the Spectral Contrastive Loss (SPL). Then, we provide relevant results from previous works on matrix factorization and contrastive learning.

2.1 Preliminaries

We assume that the data $\mathcal{D}_{orig} = \{y \mid y \sim P\}$ is sampled from a mixture distribution. The augmented data $\mathcal{D}_{aug}$ is created by applying randomly selected augmentations to each sample y. In particular, we assume that $\mathcal{D}_{aug} = \{x \mid x = y + \epsilon, y \in \mathcal{D}_{orig}\}$ where ϵ is an i.i.d. zero mean noise, such that $\mathbb{E}_\epsilon x = y$ for all x augmented from y. Let $|\mathcal{D}_{orig}| = n$ and $|\mathcal{D}_{aug}| = N$. Now, assuming the model is linear $f(x) = Wx$, where $x \in \mathbb{R}^d$ and $W \in \mathbb{R}^{d' \times d}$, the SPL in both supervised and self-supervised setting can be written as

$$\mathcal{L}_{SPL}(W) = -\text{tr}\left(W^T W C - W^T W \Sigma W^T W \Sigma\right) \tag{2.1}$$

where $\Sigma = \hat{\mathbb{E}}_{x \sim \mathcal{D}_{aug}}\left[xx^T\right]$ and C depends on the setting used. In the case of matrix factorization, it is generally assumed that $d' \leq d$, but here, there is no such assumption. For

the self-supervised setting, the matrix C can be written as $C = \hat{\mathbb{E}}_{y \sim \mathcal{D}_{orig}} \left[yy^T \right]$ due to the i.i.d. noise assumption. For the supervised setting, if denote the sample mean within each class $c \in \mathcal{C}$ by $\hat{\mu}_c$, then $C = \hat{\mathbb{E}}_c \left[\hat{\mu}_c \hat{\mu}_c^T \right]$. This trace form of the loss is useful for writing the gradient with respect to W. However, to connect it to the matrix factorization problem, we can rewrite it as

$$\mathcal{L}_{SPL}(W) = \hat{\mathbb{E}}_{x,x' \sim \mathcal{D}_{aug}} \left[(x^T W^T W x' - a_{x,x'})^2 \right] + const. \tag{2.2}$$

where $a_{x,x'} = 1$ if $x \sim x'$ and 0 otherwise. $x \sim x'$ means the two samples are similar, i.e. they have the same label in the supervised setting, and, they are are generated from the same original sample in the self-supervised setting. Since $x^T W^T W x$ is a scalar, we have the relation $x^T W^T W x = \operatorname{tr}\left(x x^T W^T W \right)$ due to the permutation invariance of the trace operation. Then, we have

$$\mathcal{L}_{SPL}(W) = \sum_i \left(\operatorname{tr}\left(A_i W^T W \right) - a_{x_{m(i)}, x_{l(i)}} \right)^2 + const. \tag{2.3}$$

where $m(i), l(i)$ maps the vectorized index i to the 2-dimensional indices m, l, and the detection matrices are given by $A_i = x_{m(i)} {x_{l(i)}}^T$. This establishes the equivalence of SPL to the matrix sensing loss up to a constant. Then, the convexity of this loss with respect to $W^T W$ directly follows.

2.2 Related Work

2.2.1 Classification

For classification tasks, the implicit bias of GD is relatively well-understood for certain loss functions. Ji et al. (2020) show that for a loss function that is convex, differentiable, and strictly decreasing to 0, which is common in classification tasks, the weights w_t updated

according to GD converge in direction to l_2-norm biased solutions. Formally, they prove

$$\lim_{t\to\infty} \frac{w_t}{\|w_t\|_2} = \lim_{B\to\infty} \frac{\bar{w}(B)}{B} \tag{2.4}$$

where

$$\bar{w}(B) = \operatorname*{argmin}_{\|w\|\leq B} \mathcal{L}(w) \tag{2.5}$$

Similarly, Sun et al. (2022) prove a similar result for the mirror descent algorithm, which generalizes the previous theoretical analysis by Ji et al. (2020).

2.2.2 Matrix factorization

For symmetric matrix factorization where the detection matrices A_i commute, Gunasekar et al. (2017) show that the gradient flow converges to the minimum $\|W^T W\|_*$ solution.

Theorem (Gunasekar et al. (2017) Theorem 1)**.** *If* $\{A_i\}_{i=1}^m$ *commute, given that the limit* $\hat{W} = \lim_{\alpha\to 0} W_\infty(\alpha\mathbb{I})$ *exists and is a global optimum for Eq. 1.5 with* $\operatorname{tr}\left(A_i \hat{W}^T \hat{W}\right) = y_i$ *for all* i*, then* $\hat{W} = \arg\min_{W\succeq 0} \|W^T W\|_*$ *subject to* $\operatorname{tr}\left(A_i W^T W\right) = y_i$.

In the more general setting, they also conjecture that this theorem holds without the commutation condition on $\{A_i\}_{i=1}^m$.

Li et al. (2021) give counter-examples to refute this conjecture, and they show that for infinitesimal initialization, given loss $\mathcal{L}(W) = f(W^T W)$ where f is convex and $\mathcal{C}^3-$smooth, the gradient flow learns the eigenvalues sequentially, from largest to smallest. This behavior can also be seen from the previous toy model in 3.1, where the limiting distributions of the time evolution of the eigenvalues of U are step functions.

To analyze the effects of depth in matrix factorization, Arora et al. (2019) first derive equations for the time evolution of the singular values and singular vectors of the product weight matrix $W = W_1 W_2 \dots W_N$ for an $N-$layer linear model $x \mapsto Wx$, then, under constant singular vectors assumption, compare the rate of different singular values as a

function of number of layers N. In particular, they show that under certain initialization assumption of $W_i(0)$,

Theorem 1 (Arora et al. (2019) Theorem 3). *If* ${W_i}^T(0)W_i(0) = W_{i+1}(0){W_{i+1}}^T(0)$ *for all* $i \in [N-1]$, *then the singular values* $\sigma_r(t)$ *of the product matrix* $W(t) = W_1(t)W_2(t)\dots W_N(t)$ *satisfy*

$$\sigma_r'(t) = -N\left(\sigma_r^2(t)\right)^{1-\frac{1}{N}}\left\langle \nabla_W l(W), u_r(t){v_r}^T(t)\right\rangle \tag{2.6}$$

where $W_i(t)$ *evolves according to*

$$\frac{dW_i(t)}{dt} = -\frac{\partial l(W)}{\partial W_i} \tag{2.7}$$

for all $i \in [N]$.

They also illustrate that increasing N increases the gaps between the learned singular values.

In the setting of a deep linear model, Gissin et al. (2020) derive exact solutions for scalar-valued weights, i.e. $W_i \in \mathbb{R}^{1\times 1}$. They generalize the scalar-valued weights to diagonalized weights and provide exact solutions for the diagonal entries.

2.2.3 Global Minima of SPL

It would be helpful to know what the global minima of the SPL looks like for a linear model when analyzing the gradient flow equations. Therefore, we provide a result from Xue et al. (2023) that looks at the minimum Frobenius norm minima of the loss.

Theorem (Xue et al. (2023) Lemma B.2). *The minimum norm minimizer, denoted by*

$$W^{**} = \arg\min_{W^*} \|W^*\|_F \quad s.t. \quad W^* \in \arg\min_{W} \mathcal{L}_{SPL}(W) \tag{2.8}$$

satisfies

$${W^{**}}^T W^{**} = \Sigma^\dagger C \Sigma^\dagger \tag{2.9}$$

for linear encodings $h_W(x) = Wx$.

Here $\Sigma^\dagger$ denotes the Moore-Penrose inverse of Σ.

Chapter 3

Results

Here, we give exact solutions to the gradient flow equation for a linear 1-layer model under certain diagonalization assumptions. Then, we give a convergence result in the Frobenius norm for a linear 1-layer model in the general setting in terms of the initial values of the weight matrix.

3.1 Toy Model

Here, we look at a 1-layer linear model without normalization under diagonalizing assumptions. Writing down the gradient flow for SPL,

$$\frac{dW}{dt} = 4WC - 4W\Sigma W^T W\Sigma \tag{3.1}$$

Symmetrizing the equation with $U(t) = W^T(t)W(t)$, we get the following expression

$$\frac{1}{4}\frac{dU}{dt} = UC + CU - U\Sigma U\Sigma - \Sigma U\Sigma U \tag{3.2}$$

Assumption 1. *The matrices C and Σ commute.*

By assumption, since both C and Σ are positive semi-definite matrices, they are simulta-

neously diagonalizable. In this model, we assume $U(0)$ is also diagonalizable by the same basis. Then, the derivatives of the non-diagonal entries in the equation stay 0 for all t and the matrix-valued ODE can be reduced to scalar-valued ODEs given by the diagonal entries. Let $\gamma_r(t)$ be the eigenvalues of $U(t)$, then we can write the following equation for all r.

$$\gamma_r'(t) = 8\gamma_r(t)\left(c_r - s_r^2\gamma_r(t)\right) \tag{3.3}$$

where c_r and s_r are the eigenvalues of C and Σ respectively. Here, we can see that since $c_r, s_r \geq 0$, for $W(t)$ to converge, we need to have $c_r = 0$ for all r where $s_r = 0$. In the general setting, this corresponds to needing $\text{colspace}\,(C) \subseteq \text{colspace}\,(\Sigma)$, which we will analyze in more detail in the next section. These equations are analytically solvable, and the solutions as a function of the initial conditions $\gamma_r(0)$ are given by the following expression.

$$\gamma_r(t) = \frac{\gamma_r(0)c_r}{\gamma_r(0)s_r^2 + (c_r - \gamma_r(0)s_r^2)e^{-8c_rt}} \tag{3.4}$$

Then,

$$\gamma_r(t) \to \begin{cases} \dfrac{c_r}{s_r^2} & c_r > 0 \\ \gamma_r(0) & c_r = 0 \end{cases} \tag{3.5}$$

Moreover, following the argument from Li et al. (2021), we can write the infinitesimal limit,

$$\lim_{\gamma_r(0)\to 0} \gamma_r\left(\frac{1}{8c_r} + \epsilon\right) \log \frac{1}{\gamma_r(0)} = \mathbb{1}_{\epsilon>0}\frac{c_r}{s_r^2} \tag{3.6}$$

which looks like a step function with respect to ϵ. This property is formalized by Gidel et al. (2019) by letting $\gamma_r(0) = \alpha$ for all r and scaling the time with $t \to -t/\log\alpha$. This makes $\epsilon = t - \frac{1}{8c_r}$ with this scaled time and the function becomes step-wise with respect to t.

Here, in the case of supervised SPL, the eigenvalues of C, c_r, are related to the l_2-norms

of the class means. In the supervised setting, the matrix C is given by

$$C = \hat{\mathbb{E}}_{c \in C} \left(\left[\hat{\mathbb{E}}_{x \in S_c} x \right] \left[\hat{\mathbb{E}}_{x \in S_c} x \right]^T \right) = \hat{\mathbb{E}}_{c \in C} \left[\hat{\mu}_c \hat{\mu}_c^T \right] \tag{3.7}$$

If the unit vectors of the classes μ_c are orthogonal, then we have exactly $c_r = \|\mu_r\|_2^2$. This shows that the classes with higher l_2-norms are learned faster. We also have, $\text{rank}\,(U) = \text{rank}\,(C) \leq |C|$. This rank condition is required for the class collapse to happen, however, it's not sufficient in general. The class collapse phenomena is defined in the following way.

Class collapse We can divide the class collapse into two main steps

1. Within-class variability collapses to zero as mappings converge to the class mean.
2. Class means converge to the vertices of a simplex on the hypershere as shown in Fig. 3.1.

Sometimes, the first part is called alignment within a specific class distribution. For example, Xue et al. (2023) show that the encodings have zero alignment with sub-classes to prove collapse to the main class mean.

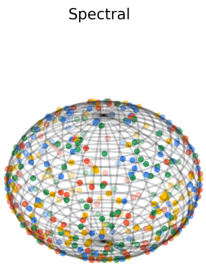
3.2 Convergence in Norm

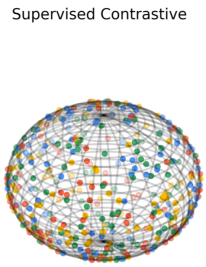
We begin by proving the following lemma.

Lemma 1. *For both supervised and self-supervised settings, we have the following relation*

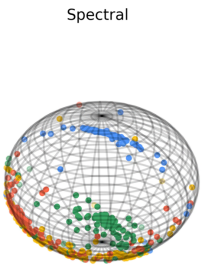
$$\text{colspace}\,(C) \subseteq \text{colspace}\,(\Sigma) \tag{3.8}$$

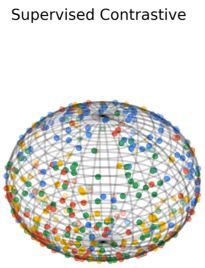
Proof. Let $C = \hat{\mathbb{E}}_c \left[\mu_c {\mu_c}^T \right]$. Then, given $x \in S_c$, we can write $x = \mu_c + \eta$ such that $\sum \eta = 0$.



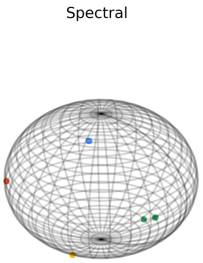


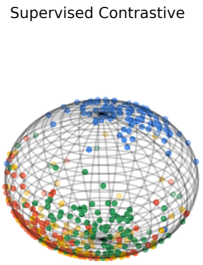
(a) Initial distributions with random labels.



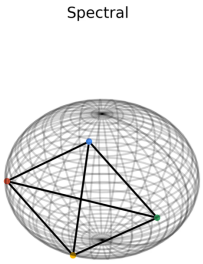


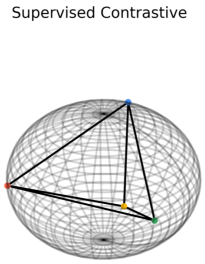
(b) InfoNCE loss starts out slow.





(c) SPL converges to class means.





(d) Final simplex structures for the two losses with 4 classes.

Figure 3.1: Following Graf et al. (2021), showing class collapse phenomena for both SPL (left) and InfoNCE (right) loss for a toy model with 4 classes.

Then, we have

$$\Sigma = \frac{1}{N} \sum_{c \in \mathcal{C}} \sum_{\eta} (\mu_c + \eta)(\mu_c + \eta)^T \tag{3.9}$$

$$= \frac{1}{N} \left(\sum_{c \in \mathcal{C}} \sum_{\eta} \mu_c {\mu_c}^T + \sum_{c \in \mathcal{C}} \sum_{\eta} \eta \eta^T \right) \tag{3.10}$$

Here, the matrix $K \coloneqq \frac{1}{N} \sum_{c \in \mathcal{C}} \sum_{\eta} \eta \eta^T$ is positive semi-definite. Then, we can write (3.10) as $\Sigma = C + K$. Now, we want to show that if Σ, C, and K are PSD, then $\text{nullspace}\,(\Sigma) \subseteq \text{nullspace}\,(C)$.

Let $u \in \mathbb{R}^d$ and assume $u \in \text{nullspace}(\Sigma)$, so $u^T \Sigma u = 0$. Then, we have

$$u^T C u + u^T K u = 0 \implies u^T C u = 0 \tag{3.11}$$

since K is PSD. Therefore $u \in \text{nullspace}(C)$. Finally, we can conclude using the rank-nullity theorem.

For $C = \hat{\mathbb{E}}_y \left[yy^T \right]$, the steps follow similarly by writing $x = y + \epsilon$ where ϵ are the 0-mean augmentations. □

Using this relation between C and Σ, and the theorem from 2.2.3, we can show that the following result holds.

Theorem 2. *Assume that the gradient flow with initial value $W(0)$ converges $W(t) \to W_\infty$, and let W^{**} be the minimum norm solution, then*

$$\|{W_\infty}^T W_\infty - {W^{**}}^T W^{**}\|_F \leq \|W(0)\|_F \left(\|W(0)\|_F + 2\|W_\infty\|_F \right) \tag{3.12}$$

Proof. Let V be an eigenbasis of Σ such that in this basis, we can write it in the following block form

$$\Sigma_V = V^{-1} \Sigma V = \begin{bmatrix} D_\Sigma & \\ & 0 \end{bmatrix} \tag{3.13}$$

By Lemma 1, we can also write

$$C = \begin{bmatrix} \tilde{C} & \\ & 0 \end{bmatrix} \tag{3.14}$$

where $\tilde{C}$ is some PSD matrix and we drop the subscript V that is used to denote we are in the basis V to make the notation simpler. Let $D_\Sigma \in \mathbb{R}^{k \times k}$, then let

$$W(t) = \begin{bmatrix} W_1(t) & W_2(t) \end{bmatrix} \tag{3.15}$$

where $W_1 \in \mathbb{R}^{d' \times k}$ and $W_2 \in \mathbb{R}^{d' \times d-k}$. Now, the gradient flow equation becomes

$$\dot{W} = W \left(\begin{bmatrix} \tilde{C} & \\ & 0 \end{bmatrix} - \begin{bmatrix} D_\Sigma & \\ & 0 \end{bmatrix} U \begin{bmatrix} D_\Sigma & \\ & 0 \end{bmatrix} \right) \tag{3.16}$$

where $U = W^T W$ as defined previously. Here, we have

$$\begin{bmatrix} D_\Sigma & \\ & 0 \end{bmatrix} U \begin{bmatrix} D_\Sigma & \\ & 0 \end{bmatrix} = \begin{bmatrix} D_\Sigma {W_1}^T(t) W_1(t) D_\Sigma & \\ & 0 \end{bmatrix} \tag{3.17}$$

Therefore, we can decouple the evolution of W_1 and W_2.

$$\dot{W}_1(t) = W_1 \left(\tilde{C} - D_\Sigma {W_1}^T(t) W_1(t) D_\Sigma \right) \tag{3.18}$$

$$\dot{W}_2(t) = 0 \tag{3.19}$$

Therefore $W_2(t) = W_2(0)$ for all t, and the gradient flow equation for $W_1(t)$ is equivalent to minimizing SPL for full rank Σ and $W \in \mathbb{R}^{d' \times k}$. By Theorem 2.2.3, all global minimizers must satisfy $W^T(\infty)W(\infty) = {D_\Sigma}^{-1} \tilde{C} {D_\Sigma}^{-1}$, therefore, if W_1 converges, we must have ${W_1}^T(\infty)W_1(\infty) = {D_\Sigma}^{-1} \tilde{C} {D_\Sigma}^{-1}$. We can write this as ${W_1}^T(\infty)W_1(\infty) = \Sigma^\dagger C \Sigma^\dagger$ in this basis. Therefore

$${W_1}^T(\infty)W_1(\infty) = {W^{**}}^T W^{**} \tag{3.20}$$

Then

$$\|{W_\infty}^T W_\infty - {W^{**}}^T W^{**}\|_F = \left\| \begin{bmatrix} 0 & {W_1}^T(\infty) W_2(0) \\ {W_2}^T W_1(\infty) & {W_2}^T(0) W_2(0) \end{bmatrix} \right\|_F \tag{3.21}$$

The Frobenius norm doesn't depend on the choice of basis, therefore this also holds in the original basis. Then, we can apply a loose upper bound based on Cauchy-Schwartz inequality

on this expression to get

$$\|{W_\infty}^T W_\infty - {W^{**}}^T W^{**}\|_F \leq 2\|W_1(\infty)\|_F \|W_2(0)\|_F + \|W_2(0)\|_F^2 \tag{3.22}$$

$$\leq \|W(0)\|_F \left(2\|W_\infty\|_F + \|W(0)\|_F\right) \tag{3.23}$$

where the last inequality follows from replacing W_1 and W_2 with W. □

As a direct consequence of Thm. 2, we can write the following statement.

Corollary 1. *As $W(0) \to 0$, ${W_\infty}^T W_\infty \to {W^{**}}^T W^{**}$ with respect to the $\|\|_F$ metric.*

3.3 Discussion

The result from Theorem 2 shows that the learned weights are close to the minimum norm minimizer of the loss function. However, this is only partly due to the bias of the gradient flow since the proof mainly relies on Lemma 1, which is a property of the matrices C and Σ. Additionally, this result is only the case for a 1-layer linear model as even in the multi-layer linear case, the proof fails to work. For large initializations, in addition to the upper bound, we can give a lower bound that makes the distance between the learned weights and the minimum norm solution arbitrarily large in the Frobenius norm by lower bounding $\|W_2(0)\|_F$ in Eq. 3.21. This is not desirable since the initialization has a big impact on the learned solutions and might be avoided with multi-layer networks, which would be an interesting direction for future work. However, for this particular setting, we can conclude that for small initializations, the learned weights with gradient flow cause class collapse in the representation space by using the the result from 2.2.3.

Chapter 4

Time Evolution of Singular Values with Depth

Here, we look at a linear N-layer deep network where the representations are given by $h_W(x) = Wx$, same as the 1-layer case. However, instead of optimizing the loss with respect to the matrix W, we have $W = W_1 W_2 \dots W_N$, and the loss is optimized with respect to the collection $\{W_i\}_{i=1}^N$. We denote the singular values of $W(t)$ by $\sigma_r(t)$ and the eigenvectors of $W^T(t)W(t)$ by $v_r(t)$. We let $W_i \in \mathbb{R}^{d_{i-1} \times d_i}$ where $d_N = d$ and $d_0 = d'$.

4.1 Singular Values of W

For the loss $\mathcal{L}_{SPL}(W_i)$, we can apply Thm.1 by Arora et al. (2019) to write down the differential equations for the singular values.

Corollary 2. *Given the assumptions in Thm.1 hold, the singular values $\sigma_r(t)$ satisfy*

$$\sigma_r'(t) = 4N \left(\sigma_r(t)\right)^{3-\frac{2}{N}} {v_r}^T(t) \left(C - \Sigma U(t) \Sigma\right) v_r(t) \tag{4.1}$$

Here, we will not be looking at the evolution of the eigenvectors $v_r(t)$, therefore, we need to use bounds on the relevant expressions. First, let $\delta > 0$ and ${v_r}^T(t) C v_r(t) \in [c_r - \delta, c_r + \delta]$

for all $t \geq T_r$, and we define $c_r \coloneqq {v_r}^T(\infty)Cv_r(\infty)$ and $T_r > 0$ such that $\sigma_r(T_r) \leq K\sigma_r(0)$ for some constant K that doesn't depend on t or r. We also need the following statement to get rid of $v_r(t)$ in Eq. 4.1.

Lemma 2. *Given A is a constant positive-definite matrix and $B(t)$ is positive semi-definite with normalized eigenvectors $u_k(t)$ and eigenvalues $b_k(t)$, then*

$$ {u_k}^T(t)AB(t)Au_k(t) \geq \gamma_{\min}(A)^2 b_k(t) \tag{4.2} $$

and if $m = \arg\max_k b_k(t)$, then

$$ {u_m}^T(t)AB(t)Au_m(t) \leq \gamma_{\max}(A)^2 b_m(t) \tag{4.3} $$

Proof. Let $Au_k = \alpha u_k + \beta(u_k)_\perp$, then ${u_k}^T B(u_k)_\perp = 0$ since u_k are the eigenvectors of B. Therefore,

$$ {u_k}^T(t)AB(t)Au_k(t) = \alpha^2 b_k + \beta^2 {(u_k)_\perp}^T B(u_k)_\perp \geq \alpha^2 b_k \tag{4.4} $$

where the last inequality follows from PSD property of B. Finally, since $\alpha = {u_k}^T Au_k \geq \gamma_{\min}(A)$ by definition, we can conclude the first inequality. For the second inequality, with the same notation we used so far, we have

$$ {(u_m)_\perp}^T B(u_m)_\perp \leq b_m \tag{4.5} $$

and therefore

$$ {u_m}^T(t)AB(t)Au_m(t) \leq (\alpha^2 + \beta^2)b_m \tag{4.6} $$

Since $(Au_k)^T Au_k = \alpha^2 + \beta^2 \leq \gamma_{\max}(A)^2$, we can conclude the second inequality. □

Proposition 1. *The maximum singular value $\sigma_m(t)$ where $m = \arg\max_k \sigma_k(t)$ satisfies*

$$ f_N(c_r - \delta, \gamma_{\max}(\Sigma)^2, t) \leq \sigma_m(t) \leq f_N(c_r + \delta, \gamma_{\min}(\Sigma)^2, t) \tag{4.7} $$

for all t such that $\sigma_m(t) > K\sigma_m(0)$, i.e. for all non-infinitesimal $\sigma_m(t)$. In addition, all singular values $\sigma_r(t)$ satisfy

$$\sigma_r(t) \leq f_N(c_r + \delta, \gamma_{\min}(\Sigma)^2, t) \tag{4.8}$$

Here, $f_N(a_r, b_r, t)$ is defined such that $f_N(a_r, b_r, 0) = \sigma_r(0)$ and

$$\int_{f_N(a_r,b_r,0)}^{f_N(a_r,b_r,t)} \frac{1}{x^{3-\frac{2}{N}}(a_r - b_r x^2)} dx = 4Nt \tag{4.9}$$

and $f_N(a_r, b_r, t) \to \sqrt{\frac{a_r}{b_r}} 1_{t > \frac{b_r}{8{a_r}^2}}$ as $N \to \infty$ if $f_N(a_r, 0) \propto \alpha/N$ for $\alpha > 0$ small.

Proof Sketch. By Lemma 2, we have the upper bounds

$$\sigma_r'(t) \leq 4N(\sigma_r(t))^{3-\frac{2}{N}} \left(c_r + \delta - \gamma_{\min}(\Sigma)^2 \sigma_r^2(t)\right) \tag{4.10}$$

Then, Eq. 4.8 directly follows. For the maximum singular value, we also have the lower bound

$$4N(\sigma_m(t))^{3-\frac{2}{N}} \left(c_m - \delta - \gamma_{\max}(\Sigma)^2 \sigma_m^2(t)\right) \leq \sigma_m'(t) \tag{4.11}$$

which gives Eq. 4.7. Finally, the limit of f_N as $N \to \infty$ follows from the properties of the hypergeometric function ${}_2F_1$ which is the solution of the integral in Eq. 4.9. □

This shows that for large N, the maximum singular value is learned at some time within $t \in [\frac{\gamma_{\min}(\Sigma)^2}{{c_m}^2}, \frac{\gamma_{\max}(\Sigma)^2}{{c_m}^2}]$, if $\delta \ll c_m$. We can restrict the range even further if we assume certain properties of Σ. In particular, if we assume that the input is batch normalized with a single batch for the full dataset, the eigenvalues of Σ can be shown to be supported on the interval $[1-\epsilon, 1+\epsilon]$ with high probability for some constant $1 > \epsilon > 0$. This way, we can write $t \in [\frac{1-\epsilon}{{c_m}^2}, \frac{1+\epsilon}{{c_m}^2}]$ with high probability.

For all singular values of $W(t)$, if N is large, we see that they stay infinitesimal up to $t_r = \frac{\gamma_{\max}(\Sigma)^2}{{c_r}^2}$, which is expected due to the sequential learning properties of factorization problems

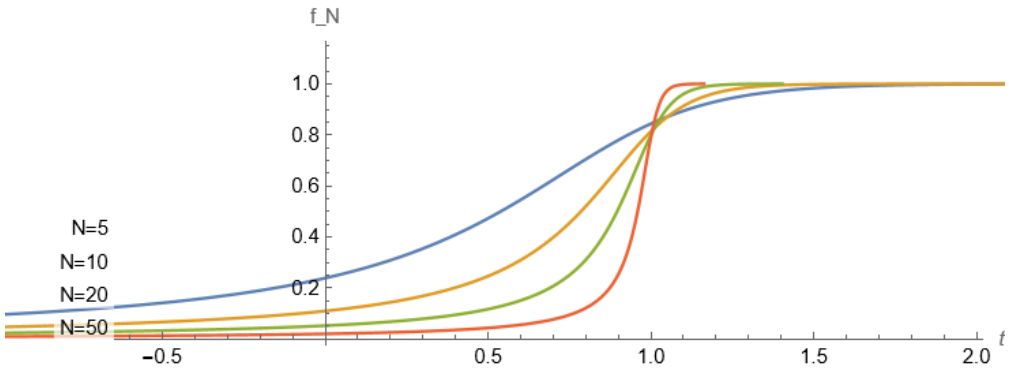
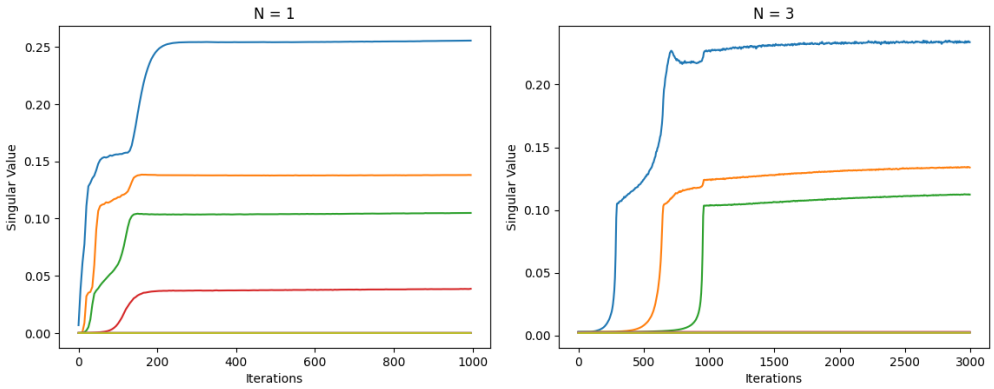


Figure 4.1: Convergence of f_N to step-wise function for large N.

and can be seen in Fig. In this case, the dependence on c_r shows that the eigenvectors with smaller projections ${v_r}^T C v_r$ are learned later during the gradient flow process. When v_r are also the eigenvectors of C, this reduces to the case for the toy model in 3.1 where the learning order depends on the corresponding eigenvalues of C.



(a) Linear model with 1 layer.

(b) Linear model with 3 layers.

Figure 4.2: Top 10 singular values of W. For $N = 3$, all the singular values are infinitesimal up to some finite time and the learning is more separated. The converged singular values are similar between (a) and (b) but in the case of $N = 3$ the last singular value is still not learned.

Proposition 2. *For a linear model with N layers satisfying the same assumptions as before, if $v_r(t) \in \ker(C)$ for all $t \geq T_0$ for some finite T_0, then $\sigma_r(t) \to 0$ with rate $\sigma_r(t) \propto t^{-\frac{N}{4N-2}}$.*

Proof. By Lemma 2, we have the following upper bound on the derivative of $\sigma_r(t)$

$$\sigma_r'(t) \leq -4N \left(\sigma_r(t)\right)^{3-\frac{2}{N}} \gamma_{\min} \left(\Sigma\right)^2 \sigma_r^2(t) \tag{4.12}$$

Let $f(t)$ solve the following initial value problem

$$f'(t) = -4N\gamma_{\min} \left(\Sigma\right)^2 \left(f(t)\right)^{5-\frac{2}{N}} \tag{4.13}$$

with $f(0) = \sigma_r(0)$. Then, the solution is given by

$$f(t) = \left(8\left(2N-1\right)\gamma_{\min}\left(\Sigma\right)^2 t + {\sigma_r}^{-4+\frac{2}{N}}(0)\right)^{-\frac{N}{4N-2}} \tag{4.14}$$

Since $\sigma_r(t) \leq f(t)$, we get the upper bound $\sigma_r(t) \lesssim t^{-\frac{N}{4N-2}}$. Now let $l = \arg\min_k \sigma_k(t)$, then

$$\sigma_l(t) \geq -4N \left(\sigma_l(t)\right)^{5-\frac{2}{N}} \gamma_{\max} \left(\Sigma\right)^2 \tag{4.15}$$

which gives $\sigma_l(t) \gtrsim t^{-\frac{N}{4N-2}}$, and the lower bound on the minimum singular value is a lower bound for all r, therefore we get the rate

$$\sigma_r(t) \propto t^{-\frac{N}{4N-2}} \tag{4.16}$$

□

4.2 Effects of Normalization

Usually, the representations $h_W(x)$ are mapped to the unit sphere $S^{d'-1}$ during training which affects the learning dynamics of the loss function. However, since the normalization makes the representations non-linear, we can model it by adding an extra term to the loss to get the expected l_2-norm of $h_W(x)$ to close to 1, similar to the analysis by Ziyin et al.

(2023). Therefore, we modify the loss as follows

$$\mathcal{L}_\lambda(W) = \mathcal{L}_{SCL}(W) + \lambda \left(\hat{\mathbb{E}}_{x \in \mathcal{D}_{aug}} \left[h_W(x)^T h_W(x) \right] - 1 \right)^2 \tag{4.17}$$

Now, if we let $C_\lambda(t) \coloneqq C + 2\lambda \left(1 - \operatorname{tr}\left(\Sigma W^T(t) W(t)\right)\right) \Sigma$ we have the following expression for the gradient flow of the singular values

$$\sigma_r'(t) = 4N \left(\sigma_r(t)\right)^{3-\frac{2}{N}} {v_r}^T(t) \left(C_\lambda(t) - \Sigma U(t) \Sigma\right) v_r(t) \tag{4.18}$$

Now, for infinitesimal initializations, $C_\lambda(0) \succ C$, and therefore the first few singular values are learned faster due to the projections ${v_r}^T C_\lambda v_r$ initially being larger for all r compared to the non-normalized loss function. This follows from Prop. 1, and for large enough N, assuming the singular values are learned sequentially, there will be a value $r = r_0$ where C_λ becomes negative definite for large enough choice of λ. This is under the assumption that the minima W_∞ without the normalization term has $\operatorname{tr}\left(\Sigma {W_\infty}^T W_\infty\right) > 1$. Then, the added term acts as an early cut-off such that the singular values with smaller c_r which would have been learned later in the process, are not learned at all. Next, we look at the time evolution of singular values corresponding to eigenvectors $v_r(t)$ of $U(t)$ that are in the kernel of C.

4.3 Discussion

The analyzes from the previous chapters show that the learning of the singular values are mainly controlled by the values $c_r = {v_r}^T(\infty) C v_r(\infty)$. These determine how fast the singular values are learned, and with an early cutoff as in 4.2, they specify which features are learned during the gradient flow. This result might explain the findings of Tack et al. (2020) if we look at what the $u^T C u$ values look like for features u that are out-of-distribution. The eigenspace of C is spanned by the main class means in the supervised setting, and since u is out-of-distribution, its decomposition in terms of the eigenvectors of $W^T W$ must include

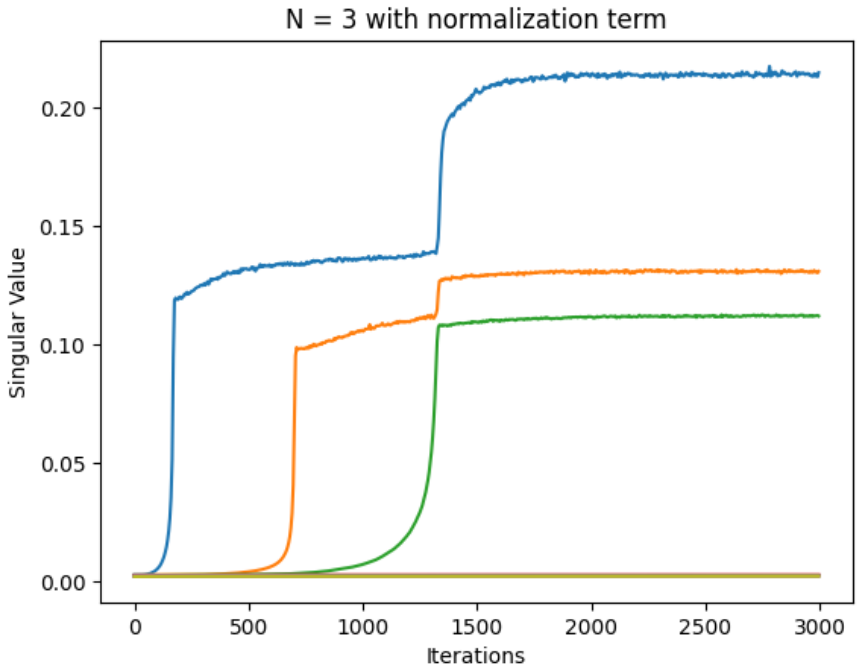


Figure 4.3: 3 layer linear model with the added normalization term. The first singular value is learned faster compared to Fig. 4.2b, but the other singular values are learned slower, increasing the separation between the highest singular value and the rest.

v_r with smaller projection values c_r on average compared to in-distribution features. Then, the lower l_2- norms of the representations of OOD data can be explained by this difference in c_r values. Additionally, the rate of convergence to 0 given in Prop. 2 show that as the number of layers N increase, the rate of convergence decreases. This is not a problem for infinitesimal initializations. However, in the general case, this result suggests that for the matrix to converge to a global minima, it requires more time as N grows larger.

Chapter 5

Future Work

In Chapter 3, we investigate the distance between the weights learned by gradient flow and the minimum norm minimizers of the loss and show that this distance depends on the initial values of the weights. However, this analyzes is limited to the 1-layer case, and a natural direction for future work on this subject would be to generalize this to deep models.

In Chapter 4, we look at the learning dynamics of the singular values. We show that the general characteristics of this dynamics can be understood by applying the results from matrix factorization which describe the sequential behavior of the learned singular values. However, in the case of SPL, in addition to the singular values, it is important to analyze the evolution and the convergence of the singular vectors to understand the alignment of the weights in terms of the distribution of the input features given by the matrices C and Σ. Most of the results here assume that the convergence of the singular vectors is on a faster scale than the singular values to get tight bounds on the time evolution path of the singular values. An interesting direction for future work would be to understand how the singular vectors evolve and to determine under what conditions this assumption holds.

Additionally, the relationship between the c_r values and the l_2-norms of the representations allows us to provide a possible explanation to the results from OOD detection by Tack et al. (2020). However, the discussion on this topic given here is too informal without

concrete statements. Therefore, another interesting direction for future work would be to make this connection more formal and provide a probabilistic description of the l_2-norms of the test samples in the representation space.

Appendix A

Properties of the Hypergeometric Function

We can use the properties of the hypergeometric function to prove the stepwise convergence of the upper and lower bounds $f_N(.)$ defined in the Prop. 1.

Proof of Prop. 1. The final part of the proof requires analyzing how the solutions $f_N(t)$ behave for large N. First, we have that the integral is given by

$$\int \frac{1}{z^{3-\frac{2}{N}}(c-az^2)}dz = z^{-2+\frac{2}{N}}\,{}_2F_1(1,-1+\frac{1}{N},\frac{1}{N},\frac{az^2}{c})\frac{N}{2c(N-1)} + conts. \tag{A.1}$$

Using the expansion of the hypergeometric function

$${}_2F_1(a,b,c;z) = \frac{\Gamma(c)}{\Gamma(a)\Gamma(b)}\sum_{n=0}^{\infty}\frac{\Gamma(n+a)\Gamma(n+b)}{\Gamma(n+c)}\frac{z^n}{n!} \tag{A.2}$$

Since $N \to \infty$, gives $c = \frac{1}{N} \to 0$ and $\Gamma(0)$ is not defined, diverges to ∞, the hypergeometric function also diverges. However, we have the well-known limit property.

$$\lim_{c\to 0}\frac{{}_2F_1(a,b,c;z)}{\Gamma(c)} = abz\,{}_2F_1(a+1,b+1,2;z) \tag{A.3}$$

Therefore,

$$\frac{{}_2F_1(1,-1+\frac{1}{N},\frac{1}{N},\frac{az^2}{c})}{\Gamma(\frac{1}{N})} \approx (-1+\frac{1}{N})\frac{az^2}{c}\,{}_2F_1(2,\frac{1}{N},2;\frac{az^2}{c}) \tag{A.4}$$

where ${}_2F_1(2,\frac{1}{N},2;\frac{az^2}{c}) = 1+O(N^{-1})$. Now, using a property of the Gamma function, we can write $\Gamma(\frac{1}{N}) = N\Gamma(1+\frac{1}{N})$, therefore for large N, $\Gamma(\frac{1}{N}) \approx N$ as $\Gamma(1+\frac{1}{N}) \to 1$. Then, we have

$${}_2F_1(1,-1+\frac{1}{N},\frac{1}{N},\frac{az^2}{c}) \approx (N-1)\frac{az^2}{c} \tag{A.5}$$

Then the integral becomes

$$N\frac{az^{\frac{2}{N}}}{2c^2} + const. = 4Nt \tag{A.6}$$

Therefore, $f_N(t) \propto \left(\frac{8c^2}{a}t\right)^{\frac{N}{2}}$ while $\left(\frac{8c^2}{a}t\right)^{\frac{N}{2}} < 1$ which is necessary for the convergence of the hypergeometric function. This shows that for $t < \frac{a}{8c^2}$, $f_N(t) \approx 0$. Additionally, this also shows that $f_N(t)$ grows fast for $t \geq \frac{a}{8c^2}$. However, we know that as $t \to \infty$, we get $f_N(t) \to \sqrt{\frac{c}{a}}$ since the only pole of the integral with $+\infty$ is at $\sqrt{\frac{c}{a}}$, and its solutions are not real for $f_N(t) > \sqrt{\frac{c}{a}}$. Therefore $f_N(t)$ will converge to the pole as $t \to \infty$, and solving for

$$\left(\frac{8c^2}{a}t\right)^N = (\frac{8c^2}{a}(\frac{a}{8c^2}+\delta))^N \approx \frac{c}{a} \tag{A.7}$$

$f_N(t)$ must grow around $t = \frac{a}{8c^2} + O(N^{-1})$. □

Bibliography

[1] Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2019.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 2020. URL http://proceedings.mlr.press/v119/chen20j.html.

[3] Tomer Galanti, András György, and Marcus Hutter. On the role of neural collapse in transfer learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=SwIp410B6aQ.

[4] Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. *Implicit regularization of discrete gradient dynamics in linear neural networks*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[5] Daniel Gissin, Shai Shalev-Shwartz, and Amit Daniely. The implicit bias of depth: How incremental learning drives generalization. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=H1lj0nNFwB.

[6] Florian Graf, Christoph Hofer, Marc Niethammer, and Roland Kwitt. Dissecting supervised contrastive learning. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 3821–3830. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/graf21a.html.

[7] Suriya Gunasekar, Blake E Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. Implicit regularization in matrix factorization. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/58191d2a914c6dae66371c9dcdc91b41-Paper.pdf.

[8] Jeff Z. HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. In A. Beygelzimer,

Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=mjyMGFL8N2.

[9] Ziwei Ji, Miroslav Dudík, Robert E. Schapire, and Matus Telgarsky. Gradient descent follows the regularization path for general losses. In Jacob Abernethy and Shivani Agarwal, editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 2109–2136. PMLR, 09–12 Jul 2020. URL https://proceedings.mlr.press/v125/ji20a.html.

[10] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf.

[11] Zhiyuan Li, Yuping Luo, and Kaifeng Lyu. Towards resolving the implicit bias of gradient descent for matrix factorization: Greedy low-rank learning. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=AHOs7Sm5H7R.

[12] Haoyuan Sun, Kwangjun Ahn, Christos Thrampoulidis, and Navid Azizan. Mirror descent maximizes generalized margin and can be implemented efficiently. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 31089–31101. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/c9694bf4f9bf3626f7d21158bab74f8e-Paper-Conference.pdf.

[13] Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. In *Advances in Neural Information Processing Systems*, volume 33, pages 11839–11852. Curran Associates, Inc., 2020.

[14] Yihao Xue, Siddharth Joshi, Eric Gan, Pin-Yu Chen, and Baharan Mirzasoleiman. Which features are learnt by contrastive learning? On the role of simplicity bias in class collapse and feature suppression. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 38938–38970. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/xue23d.html.

[15] Liu Ziyin, Ekdeep Singh Lubana, Masahito Ueda, and Hidenori Tanaka. What shapes the loss landscape of self supervised learning? In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=3zSn48RUO8M.