Benefits of Jointly Training Autoencoders: An Improved Neural Tangent Kernel Analysis

Thanh V. Nguyen\textsuperscript{c}, Raymond K. W. Wong\textsuperscript{d}, and Chinmay Hegde\textsuperscript{c}, \textit{Senior Member, IEEE}

Abstract—Deep neural networks can achieve impressive performance in the regime where they are massively over-parameterized. Consequently, over the past year, there has been a growing interest in analyzing optimization and generalization properties of over-parameterized networks. However, the majority of existing work only applies to supervised learning. The role of over-parameterization in the unsupervised setting has been contrasted against various related examples, such as weight-tied autoencoders and ReLU activation. We first provide theoretical evidence for the memorization phenomena observed in recent work using the property that infinitely wide neural networks under gradient descent evolve as linear models. We also analyze the gradient dynamics of the autoencoders in the finite-width setting. Starting from a randomly initialized autoencoder network, we rigorously prove the linear convergence of gradient descent in two weakly-trained and jointly-trained regimes. Our results indicate the considerable benefits of joint training over weak training in finding global optima, achieving a dramatic decrease in the required level of over-parameterization. Finally, we analyze the case of weight-tied autoencoders and prove that in the over-parameterized setting, training such networks from randomly initialized points leads to certain unexpected degeneracies.

Index Terms—Autoencoders, neural tangent kernel, gradient dynamics.

I. INTRODUCTION

Deep neural networks have achieved great success in a variety of applications such as image and speech recognition, natural language processing, and gaming AI. Remarkably, neural networks that achieve the state-of-the-art performance in each of these tasks are all massively over-parameterized, with far more weight parameters than the size of training data and the input dimension. Such networks can gain impressive performance in terms of both (near) zero training error and high generalization capacity, which seemingly contradicts the conventional wisdom of bias-variance tradeoffs. Surprising enough is the fact that (stochastic) gradient descent and its variants can effectively find global and generalizable solutions. Explaining this phenomenon has arguably become one of the fundamental tasks for demystifying deep learning.

As a consequence, there has been growing interest in understanding the power of the gradient descent for over-parameterized networks. Over the past year, a specific line of research [1]–[7] has led to exciting theoretical progress. In particular, the seminal work of [4] shows that gradient descent on two-layer neural networks with ReLU activation provably converges to some global minimum at a geometric rate, provided a sufficiently large number of neurons that is of polynomial order in the sample size. The key idea that leads to this result is the following: once the network is sufficiently wide, gradient descent does not change the individual weights much, but results in a non-negligible change in the network output that exponentially reduces the training loss with iteration count. This line of thinking has been subsequently refined and linked to the stability of a special kernel, called the neural tangent kernel (NTK) [8], [6] showed that the minimum eigenvalue of the limiting kernel governs both the algorithmic convergence and the generalization performance.

Despite these exciting results, the majority of existing work has focused on supervised settings and hence are limited to tasks such as classification and regression. In contrast, the role of over-parameterization in the unsupervised setting (for tasks such as reconstruction, denoising, and visualization) has gained much less attention. An early related example in unsupervised learning can be traced back to learning overcomplete dictionaries with sparse codes [9]. Another example is the problem of learning mixtures of $k$ well-separated spherical Gaussians, where [10] showed that initializing with $O(k \log k)$ centers enables expectation-maximization to correctly recover the $k$ components.

Interesting (but limited) progress has been made towards understanding over-parameterization for autoencoders, a popular class of unsupervised models based on neural networks. [11] provided an extensive study of training highly over-parameterized autoencoders using a single sample. They empirically showed that when learned by gradient descent, autoencoders with different architectures can exhibit two inductive biases: memorization (i.e., learning the constant function) and generalization (i.e., learning the identity mapping) depending on the non-linearity and the network depth. [12] showed that over-parameterized autoencoder learning is empirically biased towards functions that concentrate
around the training samples and hence exhibits memorization. [13] empirically showed that over-parameterization benefits learning in recovering generative models with single-layer latent variables (including the sparse coding model).

However, there has been a lack of theoretical evidence that supports these observations. [11] were able to prove a result for a simple one-layer linear case while [12] also proved the concentration of outputs near the training examples for a single-layer network under a data-restrictive setting. Moreover, none of the above papers have rigorously studied the training dynamics of autoencoder models. The loss surface of autoencoder training was first characterized in [14]. Subsequently, [15] proved that under-parameterized (and suitably initialized) autoencoders performed (approximate) proper parameter learning in the regime of asymptotically many samples, building upon techniques in provable dictionary learning; cf. [16], [17].

**A. Our Contributions**

In this paper, we provide the first rigorous analysis of inductive bias of gradient descent and gradient dynamics of over-parameterized, shallow (two-layer) autoencoders. To examine the inductive bias, we use an infinite-width approximation to derive the output reconstruction in terms of its input. For the gradient dynamics, we study different training schemes and establish upper bounds on the level of over-parameterization under which (standard) gradient descent, starting from randomly initialized weights, can linearly converge to global optima provided the training dataset obeys some mild assumptions. Our specific contributions are as follows:

1) First, we build upon the results by [18] to characterize the evolution of autoencoder output via linearization and infinite-width approximation. Then, we establish the inductive bias of infinite-width autoencoders trained with gradient descent and provide insights into the memorization phenomena. While our analysis is asymptotic with respect to the network width, empirical results in [11], [18] strongly suggest that similar phenomena are exhibited at finite widths as well.

2) Next, we extend the results by [4] to the setting of over-parameterized two-layer autoencoders. This involves developing a version of the NTK for multiple outputs, which can be done in a straightforward manner by lifting the kernel matrix of a single output into a higher-dimensional space via Kronecker products.

3) Next, we study the gradient dynamics of the weakly-trained\(^1\) case where the training is done only over the weights in the encoder layer. We obtain a bound on the number of hidden neurons (i.e., level of over-parameterization) required to achieve linear convergence of gradient descent, starting from random initialization, to global optimality.

4) Next, we study the gradient dynamics of the jointly-trained case where both the encoder and decoder are trained with gradient descent. Interestingly, our bound for over-parameterization required for global convergence in the jointly trained case is significantly better compared to the weakly-trained case.

5) Finally, we study a special family of autoencoders for which the encoder and decoder are weight-tied, i.e., the two layers share the same weights (this is a common architectural choice in practical applications). For the weight-tied case, we show that even without any training, \(O(d/\epsilon)\) hidden units are able to achieve \(\epsilon\)-test error where \(d\) is the input dimension. Indeed, as the number of hidden unit increases, the autoencoder approximately recovers an identity map. Since the identity map is not particularly useful in representation learning, we speculate that training of weight-tied autoencoders under over-parameterization may lead to unexpected degeneracies.

**B. Techniques**

Our analysis of gradient dynamics extends the framework of [4] for analyzing gradient descent in overparameterized neural networks using the neural tangent kernel. The special case of autoencoder networks is somewhat more complicated since we now have to deal with multiple outputs, but the use of Kronecker products enables us to derive concise NTK’s for our setting.

The work of [4] and subsequent papers typically focus on the weakly-trained case for the supervised setting where the second layer is fixed. We derive analogous bounds for the autoencoder setting. We take a step further and sharpen the analysis for the jointly-trained case. We obtain a significantly improved bound on the requisite level of over-parameterization based on two key insights:

(i) thanks to the linear decoder or the last layer, which is often the case in practical networks, the corresponding kernel is smooth, and the improved smoothness allows gradient descent to move further from the initial point; and

(ii) with this improved smoothness, we can derive a sharper characterization of the descent trajectory length in Frobenius norm rather than column-wise Euclidean norm.

With the NTK, we extend the results in [18] to the autoencoder setting and show that the reconstruction for a given input given by infinitely wide autoencoders is a linear combination of the training samples weighted by kernel scores. We therefore shed light on the memorization phenomena observed in prior work.

**II. OVERVIEW OF MAIN RESULTS**

**Notation:** We use uppercase letters to denote matrices, and lowercase for vectors or scalars. An exception is the notation \(C\) which represents a generic scalar constant, whose value can change from line to line. A vector is interpreted as a column vector by default. We denote by \(x_i \in \mathbb{R}^d\) the \(i\)th-column (or sample) of the data matrix \(X\), and \(W = [w_1, \ldots, w_m] \in \mathbb{R}^{d \times m}\) denotes a weight matrix. Whenever necessary, we distinguish between the weight vector \(w_r\) at different algorithmic steps using an explicit \(w_r(t)\) indexed by the step \(t\). For a matrix \(A = [a_1, \ldots, a_m] \in \mathbb{R}^{d \times m}\), \(\text{vec}(A) = \)
vectorizes the matrix \( A \) by stacking its columns. The symbol \( \otimes \) denotes the Kronecker product.

We use \( \mathcal{N}(\cdot) \) and \( \text{Unif}(\cdot) \) to denote the Gaussian and uniform distributions respectively. We simply write \( \mathbb{E}_w \) instead of \( \mathbb{E}_{w \sim \mathcal{N}(0, I)} \) for brevity. Throughout the paper, we refer to an arbitrary \( \delta \) as the failure probability of some event under consideration.

A. Two-Layer Autoencoders

Our goal is to understand the inductive bias and the dynamics of learning two-layer autoencoders with gradient descent. We focus on the two-layer autoencoder architecture with the rectified linear unit (ReLU), defined by \( \phi(z) = \max(z, 0) \) for any \( z \in \mathbb{R} \). In below, when \( \phi \) is applied to a vector or a matrix, the ReLU function is applied element-wisely. Given an input sample \( x \in \mathbb{R}^d \), the autoencoder returns a reconstruction \( u \in \mathbb{R}^d \) of \( x \), given by

\[
u = \frac{1}{\sqrt{md}} A \phi(W^\top x) = \frac{1}{\sqrt{md}} \sum_{r=1}^{m} a_r \phi(w_r^\top x),
\]

where \( W = [w_1, \ldots, w_m] \) and \( A = [a_1, \ldots, a_m] \) are weight matrices of the first (encoder) and second (decoder) layers respectively. We do not consider bias terms in this work. However, in principle, the bias vector for the hidden layer can be regarded as the last column of \( W \) with the last dimension of \( x \) always being 1.

Remark 1 (Choice of Scaling Factor): Notice that we have scaled the output with \( 1/\sqrt{md} \), where \( 1/\sqrt{m} \) is the factor for the first layer and \( 1/\sqrt{d} \) for the second layer. Such scaling has been utilized in mathematical analyses of supervised networks [8] as well as of autoencoders [23]. Since the ReLU is homogeneous to scaling, such factors can technically be absorbed into the corresponding weight matrices \( W \) and \( A \), but we find that keeping such factors explicit is crucial to understand the asymptotic behavior of neural network training as the network widths (i.e., \( m \) in this case) go to infinity.

Let us now set up the problem. Suppose that we are given \( n \) training samples \( X = [x_1, x_2, \ldots, x_n] \). We assume that each weight is randomly and independently initialized. Then, we train the autoencoder via gradient descent over the usual squared-error reconstruction loss:

\[
L(W, A) = \frac{1}{2} \sum_{i=1}^{n} \| x_i - \frac{1}{\sqrt{md}} A \phi(W^\top x_i) \|^2 = \frac{1}{2} \sum_{i=1}^{n} \| x_i - u_i \|^2.
\]

(II.1)

Throughout the paper, unless otherwise specified, we make the following assumptions:

Assumption 1: All training samples are normalized, i.e., \( \| x_i \| = 1 \) for \( i = 1, \ldots, n \).

We gather the training samples into the data matrix \( X = [x_1, x_2, \ldots, x_n] \) and define \( \lambda_n \triangleq \| X^\top X \|. \) Assumption 1 implies that \( \| X \|_F = \sqrt{n} \) and hence \( 1 \leq \lambda_n \leq n \). We regard \( \lambda_n \) as a parameter that depends on the data geometry. For certain families of matrices (e.g., those with independent Gaussian entries), \( \lambda_n \sim O(\max(n/d, 1)) \), which can be \( o(n) \) depending on how large \( n \) is in terms of \( d \). We note that throughout our analysis, \( X \) is regarded as fixed, and we will focus on the randomness in the weights.

Assumption 2: Consider a random vector \( w \sim \mathcal{N}(0, I) \) and define \( \bar{x}_i = 1[w^\top x_i \geq 0]x_i \) for each \( i \in [n] \). Let \( \bar{X} = [\bar{x}_1, \ldots, \bar{x}_n] \). Assume \( \min(\lambda_{\min}(\mathbb{E}_w[X^\top X]), \lambda_{\min}(\mathbb{E}_w[\phi(X^\top w)\phi(w^\top X)])) = \lambda_0 > 0 \).

The matrix \( \mathbb{E}_w[X^\top X] \) is the so-called Gram matrix from the kernel induced by the ReLU transformation and has been extensively studied in [4], [6], [20], [21]. Although this condition is difficult to interpret, one sufficient condition established in [5] (Lemma H.1 and Lemma H.2) is that as long as the squared minimum singular value \( \sigma^2_{\min}(X^\top X) > 0 \) where \( * \) denotes the Khatri-Rao product, then Assumption 2 holds. In this sense, our assumption is similar to that of [5] and slightly stronger than the one in [4], which only require \( \lambda_{\min}(\mathbb{E}_w[X^\top X]) > 0 \).

The above assumptions about the data are relatively mild, which are in sharp contrast with assuming a specific generative model for the data (e.g., dictionary models, mixture of Gaussians [13], [15]) that have so far been employed to analyze autoencoder gradient dynamics.

B. Learning Dynamics

Depending on which weight variables are being optimized, we consider three training regimes:

- **Weakly-trained case**: This corresponds to the regime where the loss function (II.1) is optimized over the weak weights \( W \) while keeping \( A \) fixed. A different form of weak training is to fix the encoder and optimize (II.1) over \( A \). Indeed, this practice is perhaps a folklore: it corresponds to standard kernel regression where the global convergence depends on the Hessian associated with random ReLU features. We do not pursue this case any further since kernel methods are well understood, but note in passing that the Hessian will eventually show up in our analysis.

- **Jointly-trained case**: This corresponds to the regime that (II.1) is optimized over both \( W \) and \( A \). This case matches practical neural network training, and performs better than the weakly trained case. We will show that the contrast between weakly-trained and jointly-trained cases arises due to the nature of the different NTK’s and our analysis may pave the way to better understanding of autoencoder training.

- **Weight-tied case**: Weight-tying is another common practice in training autoencoders. Here, one sets the encoder and decoder weights to be the same, i.e., \( A = W \), and optimizes (II.1) over the common variables \( W \).

We study this problem from the perspective of over-parameterization and show that this case leads to somewhat unexpected degeneracies.

We adopt the framework introduced in [4]. Our proofs proceed generally as follows:
We will consider the continuous flow of the reconstructions $U(t) = [u_1(t), u_2(t), \ldots, u_n(t)] \in \mathbb{R}^{d \times n}$ corresponding to the samples in $X$ at time $t$. This continuous flow can be morally viewed as the execution of gradient descent with infinitesimal learning rate. This enables us to write:

$$\frac{d \text{vec}(U(t))}{dt} = K(t) \text{vec}(X - U(t)),$$

where $K(t)$ is a kernel matrix.

From this characterization, we can infer that the spectrum of $K(t)$ governs the dynamics of the outputs. To derive explicit convergence bounds, we will first prove that $K(0)$ has positive minimum eigenvalue with high probability. This is achieved via using concentration arguments over the random initialization. Then, we will upper-bound the movement of each individual weight vector from the initial guess and hence bound the deviation of $K(t)$ from $K(0)$ in terms of spectral norm.

By discretizing the continuous-time analysis, we will obtain analogous bounds for gradient descent with a properly chosen step size and show that gradient descent linearly converges to a global solution.

Our convergence results are informally stated in the following theorems:

**Theorem 1 (Informal Version of Theorems 4 and 5):** Consider an autoencoder that computes output $u = \frac{1}{\sqrt{md}} \text{vec}(W'x)$ where the weight vectors are initialized with independent vectors $w_r \sim N(0, 1)$ and $a_r \sim \text{Unif}([−1, 1]^d)$ for all $r \in [m]$. For any $\delta \in (0, 1)$ and $m \geq C \frac{d^2}{\lambda_0 \delta}$ for some large enough constant $C$, the gradient descent over $W$ linearly converges to a global minimizer with probability at least $1 - \delta$ over the randomness in the initialization.

**Theorem 2 (Informal Version of Theorems 6 and 7):** Consider an autoencoder that computes output $u = \frac{1}{\sqrt{vd}} \text{vec}(W'x)$ where the weight vectors are initialized with independent vectors $w_r \sim N(0, 1)$ and $a_r \sim \text{Unif}([−1, 1]^d)$ for all $r \in [m]$. For any $\delta \in (0, 1)$ and $m \geq C \frac{d^2}{\lambda_0 \delta}$ for some large enough constant $C$, the gradient descent jointly over $W$ and $A$ linearly converges to a global minimizer with probability at least $1 - \delta$ over the randomness in the initialization.

**Comparisons With Existing Work:** We summarize the quantitative implications of our results in Table I. In this table, we compare with [4], [5], [7] that achieve the best known bounds to our knowledge. Note that we do not compare with [22] because the similar bound $\Omega(n^\delta)$ in that work only applies to smooth activations. We also emphasize that the factor $d$ in our bounds arises due to the fact that our network produces high-dimensional outputs (dimension $d$ in the case of autoencoders) while the previous works have focused on scalar outputs. It is worth noting that the input dimension $d$ is implicit in $\lambda_0$ and $\lambda_n$.

For weakly-trained networks with a single output, we (slightly) improve the order of over-parameterization: $m = \Omega\left(\frac{n^4 \lambda_0}{\lambda_n}\right)$ over the previous bound $\Omega\left(\frac{n^5}{\lambda_n}\right)$ in [4, Theorem 3.2] by explicitly exposing the role of the spectral norm $\lambda_0$ of the data.

For the jointly-trained regime, we obtain a significantly improved bound over [4, Theorem 3.3]. Our result is consistent with [5, Theorem 6.3], but we have both layers jointly trained; the proof technique in [5, Theorem 6.3] is different from ours (bounding Jacobian perturbations), and does not seem to be easily extended to the jointly trained case.

Let us explain the results in Table I in terms of the dimension $d$ and the sample size $n$. We emphasize that in the fairly typical regime of machine learning where $n \geq d$ and $\lambda_n \sim n/d$, the level of over-parameterization for the single output is moderate (of order $n^4/d^4$). Since autoencoders have an output dimension $d$, the factor-$d$ in the bounds is natural in the jointly-trained case by characterizing the trajectory length by Frobenius norm. This is consistent with the result in [7]. Our bound is different from that in [7] in that we make assumption on the minimum eigenvalue $\lambda_0$ while they assume a lower bound on the sample separation $\Delta$. A direct universal comparison between the two bounds is difficult; however, [5] shows an upper bound $\lambda_0 \geq \Delta/100n^2$. Finally, we note that initializing $A$ with i.i.d. Rademacher entries keeps our analysis in line with previous work, and an extension to Gaussian random initialization of $A$ should be straightforward. Our results can be extended to deep feed-forward autoencoders where the neural tangent kernel is simply the sum of the individual kernels for each layer, and one can focus on the kernel with respect to the last layer’s weights.

### Table I

| Regime         | Reference | Single output | Multiple output |
|----------------|-----------|---------------|-----------------|
| Weakly-trained | [4]       | $C n^6 \lambda_0^4 / \lambda_n^3$ | $\times$        |
| Our work       | $C n^6 \lambda_0^4 / \lambda_n^3$ | $\times$        |
| Jointly-trained| [4]       | $C n^6 \log(n/\delta) \lambda_0^4 / \lambda_n^3$ | $\times$        |
| [7]            | $C n^6 \lambda_0^4 / \lambda_n^3$ | $\times$        |
| Our work       | $C n^6 \lambda_0^4 / \lambda_n^3$ | $\times$        |

### C. Inductive Bias

The following theorem establishes a result on the inductive bias of the infinitely wide autoencoders trained with gradient descent.

**Theorem 3:** Let $K^\infty = \mathbb{E}_{W(0), A(0)}[K(0)]$. Assume $\lambda_{\min}(K^\infty) > 0$ and let $\eta_{\text{critical}} = 2(\lambda_{\max}(K^\infty) + \lambda_{\min}(K^\infty))^{-1}$. Under gradient descent with learning rate $\eta < \eta_{\text{critical}}$, for every normalized $x \in \mathbb{R}^d$ as the width...
m \to \infty$, the autoencoder output $f_t(x)$ at step $t$ converges to $\mu_t(x) + \gamma_t(x)$, with:

$$
\mu_t(x) \to \sum_{i=1}^{n} \Lambda_i x_i,
$$

$$
\gamma_t(x) \to f_0(x) - \sum_{i=1}^{n} \Lambda_i f_0(x_i)
$$

where each matrix $\Lambda_i \in \mathbb{R}^{d \times d}$ depends on the kernel score between the input $x$ and each training sample $x_i$, and $K^\infty$. $f_0(x)$ is the initial reconstruction of the autoencoder at $x$.

We prove this result in Section III-B. Despite its asymptotic nature, Theorem 3 generalizes the simple result in [11, Theorem 1] to non-linear autoencoders and multiple-sample training. We will derive $\Lambda_i$ specifically for infinitely wide autoencoders and show that they exhibit “memorization” in the sense that the closer the new test input $x$ is to the span of training data $X$, the more its reconstruction concentrates around these seen points — an intriguing phenomenon empirically observed in [12].

III. THE NEURAL TANGENT KERNEL AND LINEARIZATION OF AUTOENCODERS

A. NTK for General Autoencoders

We first give a compact expression of the neural tangent kernels for general autoencoders (possibly deep and with more than 2 layers) with multiple outputs. Given $n$ i.i.d samples $X = [x_1, x_2, \ldots, x_n]$ and the autoencoder $f(\theta, x)$, we consider minimizing the squared-error reconstruction loss:

$$
L(\theta) = \frac{1}{2} \sum_{i=1}^{n} \|x_i - f(\theta, x_i)\|^2 = \frac{1}{2} \sum_{i=1}^{n} \|x_i - u_i\|^2,
$$

where $\theta$ is a vector that stacks all the network parameters (e.g., the weight matrices $W$ and $A$ in the aforementioned two-layer setting) and $u_i = f(\theta, x_i) \in \mathbb{R}^d$ denotes the corresponding output for every $i = 1, 2, \ldots, n$. The evolution of gradient descent on $L(\theta)$ with an infinitesimally small learning rate is represented by the following ordinary differential equation (ODE):

$$
\frac{d\theta(t)}{dt} = -\nabla_\theta L(\theta(t)). \tag{III.1}
$$

The time-dependent NTK for autoencoders can be characterized as follows:

**Lemma 1:** Denote by $U(t) = [u_1(t), u_2(t), \ldots, u_n(t)] \in \mathbb{R}^{d \times n}$ the corresponding outputs of all the samples in $X$, i.e., $u_i(t) = f(\theta(t), x_i)$. The dynamics of $U(t)$ is given by the ODE:

$$
\frac{d\text{vec}(U(t))}{dt} = K(t)\text{vec}(X - U(t)),
$$

where $K(t)$ is an $nd \times nd$ positive semi-definite kernel matrix whose $(i, j)$-th block of size $d \times d$ is:

$$
\left( \frac{\partial}{\partial \theta} f(\theta, x_i) \right) \cdot \left( \frac{\partial}{\partial \theta} f(\theta, x_j) \right) \top.
$$

If the parameters $\theta(0)$ are assumed to be stochastic, then the (deterministic) neural tangent kernel (NTK) is defined as:

$$
(K^\infty)_{i,j} = \mathbb{E}_{\theta(0)} \left[ \left( \frac{\partial}{\partial \theta} f(\theta(0), x_i) \right) \cdot \left( \frac{\partial}{\partial \theta} f(\theta(0), x_j) \right) \right] \top. \tag{III.2}
$$

Note that $K^\infty$ is time-independent. If the network is randomly initialized and its width is allowed to grow infinitely large, $K(t)$ converges to $K^\infty$, and remains constant during training. Our goal is to show that if the width is sufficiently large (not necessarily infinite), then $K(t) \approx K(0) \approx K^\infty$, and the gradient dynamics are governed by the spectrum of $K^\infty$.

B. Linearization of Autoencoders

While the NTK allows us to analyze the gradient dynamics of autoencoders, it does not provide an interpretable characterization of the reconstruction given a new input. This makes it difficult to reason about the inductive bias of the over-parameterization and gradient descent for autoencoders, which were empirically studied in [11], [12]. Our work takes a step further and theoretically characterizes these empirical results. To this end, we leverage the linearization and infinite approximation techniques in [18].

For the autoencoder $f(\theta, x)$, we denote by $\theta(t)$ the parameter vector at time $t$ and by $\theta(0)$ its initial value. We simplify the notation by letting $f_t(x) = f(\theta(t), x)$ and $f_t(X) = [f_t(x_1), \ldots, f_t(x_n)]$.

We first give a compact expression of the neural tangent kernel (NTK) is defined as:

$$
(K^\infty)_{i,j} = \mathbb{E}_{\theta(0)} \left[ \left( \frac{\partial}{\partial \theta} f(\theta(0), x_i) \right) \cdot \left( \frac{\partial}{\partial \theta} f(\theta(0), x_j) \right) \right] \top.
$$

Recall the gradient flow characterization of the dynamics on $L(\theta)$ in (III.1) and consider the following linearized autoencoder via the first order Taylor expansion of $f_t(x)$ around $\theta(0)$:

$$
f_t^{\text{lin}}(x) \triangleq f_0(x) + \frac{\partial f_0(x)}{\partial \theta} \cdot \omega(t),
$$

where $\omega(t) = \theta(t) - \theta(0)$ is the parameter movement from its initialization. The first term $f_0(x)$ or the initial reconstruction of $x$ remains unchanged during training over $\theta$ whereas the second term captures the dynamics with respect to the parameters, governed by:

$$
\dot{\omega}(t) = \sum_{i=1}^{n} \left( \frac{\partial f_0(x_i)}{\partial \theta} \right) \top (x_i - f_t^{\text{lin}}(x_i)), \tag{III.3}
$$

$$
f_t^{\text{lin}}(x) = \sum_{i=1}^{n} \left( \frac{\partial f_0(x_i)}{\partial \theta} \right) \top (x_i - f_t^{\text{lin}}(x_i)). \tag{III.4}
$$

In the above expressions, we denote

$$
\nabla_\theta f_0(X)^\top \triangleq \left[ \left( \frac{\partial f_0(x_1)}{\partial \theta} \right)^\top, \ldots, \left( \frac{\partial f_0(x_n)}{\partial \theta} \right)^\top \right]^\top,
$$

$$
K_0(x, X) \triangleq \frac{\partial f_0(x)}{\partial \theta} \nabla_\theta f_0(X)^\top \in \mathbb{R}^{d \times nd},
$$

$$
K_0 = K_0(X, X) \triangleq \nabla_\theta f_0(X)\nabla_\theta f_0(X)^\top \in \mathbb{R}^{nd \times nd}.
$$

The last quantity is known as the neural tangent kernel matrix evaluated at $\theta(0)$, which we have seen in the earlier section.
Following from [18], we obtain the closed form solutions for the ODEs in (III.3) and (III.4) as follows:

$$\omega(t) = \nabla_\theta f_0(X)^T K_0^{-1}(I - e^{-K_0 t}) \text{vec}(X - f_0^{lin}(X)), $$

(III.5)

$$\text{vec}(f_t^{lin}(X)) = (I - e^{-K_0 t}) \text{vec}(X) + e^{-K_0 t} \text{vec}(f_0(X)).$$

(III.6)

Moreover, given any new input $x$, the linearized output is $f_t^{lin}(x) = \mu_t(x) + \gamma_t(x)$ where the signal and noise terms are given by

$$\mu_t(x) = K_0(x, X)K_0^{-1}(I - e^{-K_0 t}) \text{vec}(X),$$

(III.7)

$$\gamma_t(x) = f_0(x) - K_0(x, X)K_0^{-1}(I - e^{-K_0 t}) \text{vec}(f_0(X)).$$

(III.8)

These equations characterize the dynamics of reconstruction (up to scaling) for the linearized network, which establishes the connection between the infinitely wide autoencoder and its linearized version. Now, we prove Theorem 3.

**Proof of Theorem 3**: We simply invoke Theorem 2.1 in [18] for the autoencoder case. Denote by $K_0 = [E^{W_k(0), A_0(0)} K(0)]$ the neural tangent kernel of the two-layer autoencoder. Assume $\lambda_{\text{min}}(K_0) > 0$ and let $\eta_{\text{critical}} \triangleq 2(\lambda_{\text{max}}(K_0) + \lambda_{\text{min}}(K_0))^{-1}$. [18] shows that under gradient descent with learning rate $\eta < \eta_{\text{critical}}$, for every $x \in \mathbb{R}^d$ such that $\|x\| \leq 1$, as the width $m \rightarrow \infty$, the autoencoder $f_0(x)$ converges in $f_t^{lin}(x)$ given by Equation (III.7) and Equation (III.8).

IV. **INDUCTIVE BIAS OF INFINITELY WIDE TWO-LAYER AUTOENCODERS**

In principle, the training dynamics of over-parameterized autoencoders are similar to those of supervised networks. However, the generalization properties or inductive biases of the over-parameterization are different and underexplored. In this section, we rigorously analyze the observations in [11], [12] using the results we have developed.

A. **One-Sample Training**

This training setting was exclusively studied in [11] with interesting insights on the memorization phenomenon and the role of the depth and width. They were able to give a theoretical justification for their observation in a very simple one-layer linear case. Using linearization, we generalize this result for non-linear networks. We particularly focus on the two-layer architecture, but the results can be extended to networks of any depth. Although our result is asymptotic, [18] showed that networks with finite, large width exhibit the same inductive bias.

Suppose we have access to only one sample $x$, or the training data $X = x$. For a test input $x'$, we have $f_t^{lin}(x') = \mu_t(x') + \gamma_t(x')$ where (III.7) and (III.8) gives

$$\mu_t(x') = K_0(x', x)K_0^{-1}(I - e^{-K_0 t}) x, $$

$$\gamma_t(x') = f_0(x') - K_0(x', x)K_0^{-1}(I - e^{-K_0 t}) f_0(x).$$

As the learning rate for gradient descent is sufficiently small and in the infinite-width limit $m \rightarrow \infty$, the autoencoder output $f_t(x') \rightarrow f_t^{lin}(x')$. Moreover, in this kernel regime and two-layer case, the neural tangent kernel is much simplified, which is derived in Appendix F as follows:

$$K_0 \rightarrow \frac{I}{d}, \quad K_0(x', x) \rightarrow \langle x', x \rangle \frac{\pi - \arccos \langle x', x \rangle}{\pi d} I + \frac{1}{2\pi d} \sqrt{1 - \langle x', x \rangle^2} I.$$ 

When $x'$ is close to $x$, $K_0(x', x) \sim I/d$, the signal term $\mu_t(x') \approx d \cdot K_0(x', x) x \approx x$ dominates the noise term $\gamma_t(x') \approx f_0(x') - f_0(x)$, then the reconstruction is close to $x$. When $x'$ is far from $x$, $\mu_t(x') \sim 0$ while $\gamma_t(x')$ is a random, so the reconstruction is governed by a random noise. This result is consistent with the empirical evidence in [11].

B. **Multiple-Sample Training**

For the training with many samples, [12] showed that over-parameterized autoencoders exhibit memorization by learning functions that concentrate near the training examples. They proved that single-layer autoencoders project data onto the span of the training examples. We provide another intuition based on the reconstruction of the linearized networks. For an arbitrary input $x'$, we have from (III.7) and (III.8) $f_t^{lin}(x') = \mu_t(x') + \gamma_t(x')$ where

$$\mu_t(x') = K_0(x', x)K_0^{-1}(I - e^{-K_0 t}) \text{vec}(X), $$

$$\gamma_t(x') = f_0(x') - K_0(x', X)K_0^{-1}(I - e^{-K_0 t}) \text{vec}(f_0(X)).$$

The signal part of the reconstruction is a linear combination of training samples weighted by the kernel $K_0(x', x_i)$ and the eigenvalues of the kernel matrix $K_0$. Therefore, as $m \rightarrow \infty$ and $t$ is sufficiently large,

$$\mu_t(x') \rightarrow \sum_{i=1}^{n} \Lambda_i x_i, \quad \text{and} \quad \gamma_t(x') \rightarrow f_0(x') - \sum_{i=1}^{n} \Lambda_i f_0(x_i),$$

where each $\Lambda_i \in \mathbb{R}^{d \times d}$ depends on $K_0(x', X)$ and $K_0^{-1}$. Since $\Lambda_i$ has smaller entries when $x'$ is further from the training samples in $X$ and vice versa, then one can see that the closer the new test input $x'$ is to the span of training data $X$, the more its reconstruction concentrates around these seen points. This coincides with the observation about “memorization” by [12].

V. **NTK OF TWO-LAYER AUTOENCODERS**

Since we consider the two training regimes, including the **weakly-trained** and **jointly-trained**, we first give the expression of a few base kernels whose appropriate combination produce the final kernel for each individual case. The precise kernel derivation of each regime is given in the dedicated section.

We focus on gradient descent on the squared reconstruction loss

$$L(W, A) = \frac{1}{2} \sum_{i=1}^{n} \|x_i - \frac{1}{\sqrt{n d}} A \phi(W^T x_i)\|^2$$

$$= \frac{1}{2} \sum_{i=1}^{n} \|x_i - u_i\|^2,$$
where the weights are independently initialized such that:
\[ w_r(0) \sim \mathcal{N}(0, I), \quad a_r(0) \sim \text{Unif}\{-1, 1\}^d, \quad r = 1, \ldots, m. \]
Here the minimization can be either over the encoder weights \( W \), or the decoder weights \( A \), or both \( W \) and \( A \). Let us denote
\[ \tilde{X}_t(t) = \left[ \mathbb{1}[w_r(t) \, x_1 \geq 0; x_1, \ldots, \mathbb{1}[w_r(t) \, x_n \geq 0; x_n] \right]. \]
If we fix \( A \) and optimize the loss \( L(W, A) \) over \( W \), we get
\[ G(t) = \frac{1}{md} \sum_{r=1}^{m} \tilde{X}_t(t) \, \tilde{X}_t(t) \, a_r, a_r^T. \]
If we fix \( W \) and optimize the loss \( L(W, A) \) over \( A \), we get
\[ H(t) = \frac{1}{md} \sum_{r=1}^{m} \phi(X^T \, w_r(t)) \, \phi(w_r(t) \, X) \, I, \]
Writing these kernels in Kronecker product form allows us to clearly manifest the connection to the supervised learning case, and enables characterization of their spectrum. Intuitively, in the jointly-trained case, since both \( W \) and \( A \) depend on \( t \), an invocation of the chain rule leads to the sum \( G(t) + H(t) \) being the “effective” kernel that governs the joint dynamics.

In the infinite-width limit where \( m \to \infty \), the NTKs in the corresponding training regimes reduce to compositions of the following fixed deterministic kernels:
\[
G^\infty = \mathbb{E}_{w(0), a(0)} \left[ \tilde{X}(0)^T \tilde{X}(0) \, a(0) a(0)^T \right] = \mathbb{E}_{w(0)} [\tilde{X}(0)^T \tilde{X}(0)] \otimes I, \\
H^\infty = \mathbb{E}_{w(0)} \left[ \phi(X^T \, w(0)) \, \phi(w(0) \, X) \right] \otimes I. 
\]
Somewhat curiously, we will show that the crucial component of the time-dependent kernel in the jointly-trained regime, \( H(t) \) (within \( H(t) + G(t) \)), is better-behaved than the corresponding kernel in the weakly-trained regime, \( G(t) \), thanks to its better Lipschitz smoothness, even though the respective limiting kernels are the same. This improved smoothness allows us to derive a much better bound on kernel perturbations with respect to changing weights, and this results in a significant improvement in the level of over-parameterization (Theorem 2).

Armed with this, we start the analysis of the gradient dynamics for weakly-trained and jointly-trained autoencoders with the notations summarized in Table II.

### VI. WEAKLY-TRAINED AUTOENCODERS

First, we consider the weakly-trained regime with the usual squared reconstruction loss function:
\[ L(W) = \frac{1}{2} \sum_{i=1}^{n} \| x_i - \frac{1}{\sqrt{md}} A \phi(W^T x_i) \|^2, \]
where the corresponding minimization is only performed over \( W \). After the random initialization, we keep \( A \) fixed throughout and apply gradient descent learning over \( W \) with step size \( \eta \):
\[ W(k + 1) = W(k) - \eta \nabla_w L(W(k)), \quad k = 0, 1, 2, \ldots. \]
The gradient \( \nabla_w L(W) \) of the loss over \( w_r \) for \( r = 1, 2, \ldots, m \) is derived in Appendix C as
\[ \nabla_w L = -\frac{1}{\sqrt{md}} \sum_{i=1}^{n} \mathbb{1}[w_r(t) \, x_i \geq 0; x_i] a_r^T (x_i - u_i). \]
Note that \( \phi(z) \) is differentiable everywhere except at \( z = 0 \), at which the derivative will be considered as \( 0 \). The gradient flow for each \( w_r(t) \) is characterized via the following ODE:
\[ \frac{dw_r(t)}{dt} = -\nabla_w L(W(t)). \]
Vectorizing \( \frac{dU(t)}{dt} \), we can characterize the dynamics of the reconstructions \( \tilde{U}(t) \) by
\[ \frac{d\text{vec}(U(t))}{dt} = \frac{1}{d} K_1(t) \text{vec}(X - U(t)), \]
where \( K_1(t) \) is written in the Kronecker form whose \((i,j)\)-block is of size \( d \times d \):
\[ K_1(t) = \frac{1}{m} \sum_{r=1}^{m} \tilde{X}_r(t) \, \tilde{X}_r(t) \, a_r a_r^T, \]
and \( \tilde{X}_r(t) = \left[ \mathbb{1}[w_r(t) \, x_1 \geq 0; x_1, \ldots, \mathbb{1}[w_r(t) \, x_n \geq 0; x_n] \right]. \]
At the initialization \( W(0) \) and \( A(0) \), in the limit as \( m \to \infty \), \( K_1(0) \) converges to the NTK:
\[ K_1^\infty = \mathbb{E}_{w} [\tilde{X}^T \tilde{X}] \otimes I. \]
By Assumption 2, \( \lambda_{\min}(K_1^\infty) = \lambda_{\min}(\mathbb{E}_{w} [\tilde{X}^T \tilde{X}]) = \lambda_0 > 0 \). In other words, the NTK kernel is strictly positive definite. When \( m \) is large enough prove, \( K_1(t) \approx K_1^\infty \approx K_1^\infty \). The gradient flow converges to a global minimum via the following Theorem:

#### TABLE II

| Notation | Place | Explanation |
|----------|-------|-------------|
| \( n, m, d \) | Section II | The size of training data, the size of hidden and input layers |
| \( \lambda_0 \) | Assumption 2 | The minimum eigenvalue of the NTK |
| \( L(W, A) \) | Section III | The squared reconstruction loss over the weights \( W, A \) |
| \( U = [u_1, \ldots, u_m] \) | Section III | Reconstructions of the input samples \( x_1, \ldots, x_n \) |
| \( K_1 = G \) | Sections III and VI | NTK of the weakly-trained regime |
| \( K_2 = G + H \) | Sections III and VII | NTK of the jointly-trained regime |
Theorem 4 (Linear Convergence of Gradient Flow, Weakly-Trained Regime): Suppose Assumptions 1 and 2 hold. Suppose at initialization that the weights are independently drawn such that \( w_r \sim \mathcal{N}(0, I) \) and \( a_r \sim \text{Unif}(\{-1, 1\}^d) \) for all \( r \in [m] \).

If \( m \geq C\frac{n_d^2 \lambda \delta}{\lambda_0^2} \) for a constant \( C > 0 \), then with probability at least 1 − \( \delta \),

\[
\|X - U(t)\|_F^2 \leq \exp\left(-\frac{\lambda_0 t}{d}\right)\|X - U(0)\|_F^2.
\]

The above result for gradient flow can be viewed as a convergence rate for gradient descent in the weakly-trained regime with infinitesimally small step size. We now state the convergence result for gradient descent with finite step sizes.

Theorem 5: Suppose Assumptions 1 and 2 hold. The initial weights are independently drawn such that \( w_r \sim \mathcal{N}(0, I) \) and \( a_r \sim \text{Unif}(\{-1, 1\}^d) \) for all \( r \in [m] \). If \( m \geq C\frac{n_d^2 \lambda \delta}{\lambda_0^2} \) for some large enough constant \( C \), then with probability at least 1 − \( \delta \), the gradient descent on \( W \) with step size \( \eta = \Theta(\frac{1}{md \lambda_0}) \),

\[
\|X - U(k)\|_F^2 \leq \left(1 - \frac{\eta_0}{2d}\right)^k \|X - U(0)\|_F^2
\]

for \( k = 0, 1, \ldots \).

Theorem 5 shows the linear convergence of gradient descent to a global minimum from a random initialization. We achieve this with the number of hidden neurons \( m = \Omega\left(\frac{n^2 \lambda_0^2}{\lambda \delta^2}\right) \), which is polynomial in terms of \( n \) and \( d \). This mirrors the result for weakly-trained networks with a single output in [4, Theorem 3.2] where we (slightly) improve the order of over-parameterization: \( m = \Omega\left(\frac{n^2 \lambda_0^2}{\lambda \delta^2}\right) \) over the previous bound \( \Omega\left(\frac{n^2 \lambda_0^2}{\lambda \delta^2}\right) \) by explicitly exposing the role of the spectral norm \( \lambda_0 \) of the data. The dependence on \( d \) reflects in the factor \( 1/d \) in the step size \( \eta = \Theta(\frac{1}{md \lambda_0}) \) and the convergence rate. We give the proofs of Theorem 4 and Theorem 5 in Appendix C.

VII. JOINTLY-TRAINED AUTOENCODERS

Similarly, we analyze the jointly-trained regime where the loss is optimized over both sets of layer weights. We consider the same squared loss as the previous regime with the weights \( W \) and \( A \) initialized in the same way. The difference is that the optimization is now taken over both weights \( W \) and \( A \).

Gradient descent updates jointly on \( W \) and \( A \) with step size \( \eta \) are

\[
W(k + 1) = W(k) - \eta \nabla_W L(W(k), A(k)), \quad k = 0, 1, \ldots
\]

\[
A(k + 1) = A(k) - \eta \nabla_A L(W(k), A(k)), \quad k = 0, 1, \ldots
\]

The gradients of the squared loss over \( w_r \) and \( a_r \) are derived in Appendix D and given by

\[
\nabla_{w_r} L(W, A) = -\frac{1}{\sqrt{md}} \sum_{i=1}^{n} \mathbb{I}[w_r^T x_i \geq 0] x_i a_r (x_i - u_i),
\]

\[
\nabla_{a_r} L(W, A) = -\frac{1}{\sqrt{md}} \sum_{i=1}^{n} \phi(w_r^T x_i) (x_i - u_i).
\]

Consider two ODEs, one for each weight vector over the continuous time \( t \):

\[
\frac{dw_r(t)}{dt} = -\nabla_{w_r} L(W(t), A(t)),
\]

\[
\frac{da_r(t)}{dt} = -\nabla_{a_r} L(W(t), A(t)).
\]

The dynamics of the reconstructions \( U(t) \) are as follows:

\[
\frac{d\text{vec}(U(t))}{dt} = \frac{1}{d} \left( G(t) + H(t) \right) \text{vec}(X - U(t)).
\]

In the above equation, \( G(t), H(t) \in \mathbb{R}^{nd \times nd} \), which we introduced in Section III, are

\[
G(t) = \frac{1}{m} \sum_{r=1}^{m} \tilde{X}_r(t)^T \tilde{X}_r(t) \circ a_r(t) a_r(t)^T,
\]

\[
H(t) = \frac{1}{m} \sum_{r=1}^{m} \phi(X^T w_r(t)) \phi(w_r(t)^T X) \otimes I,
\]

where \( \tilde{X}_r(t) = [\mathbb{I}[w_r(t)^T x_1 \geq 0] x_1, \ldots, \mathbb{I}[w_r(t)^T x_n \geq 0] x_n] \).

Let us emphasize again that \( G(t) \) is precisely the kernel that governs the dynamics for the weakly-trained case. On the other hand, \( H(t) \) is a Kronecker form of the Hessian of the loss function derived with respect to \( A \), using the features produced at the output of the ReLU activations.

As shown in Section III, assuming randomness and independence of \( W(0) \) and \( A(0) \), we can prove that as \( m \rightarrow \infty \), \( H(0) \) and \( G(0) \) converge to the corresponding NTKs whose minimum eigenvalues are assumed to be positive.

Denote the time-dependent kernel \( K_2(t) = G(t) + H(t) \). Since both \( G(t) \) and \( H(t) \) are positive semi-definite, we only focus on \( H(t) \) for reasons that will become clear shortly. Since \( G(t) \) is also positive definite with high probability (Section C-B), the flow convergence can be also boosted by the positive definiteness of \( G^\infty \). By Assumption 2,

\[
\lambda_{\min}(K_2^\infty) \geq \lambda_{\min}(H^\infty) \geq \lambda_0 > 0.
\]

Since \( G(0) \) is positive semi-definite, in order to bound the minimum eigenvalue of \( K_2(0) \), all we need is to bound that of \( H(0) \). Importantly, we observe that the smoothness of the kernel \( H(t) \) is much better as a function of the deviation of the weights from the initialization. This allows the weights to change with a larger amount than merely using \( G(t) \), and enables us to significantly reduce the number of parameters required for the gradient to reach a global optimum.

Our main result for gradient flow of the jointly-trained autoencoder is given by:

Theorem 6 (Linear Convergence of Gradient Flow, Jointly-Trained Regime): Suppose Assumptions 1 and 2 hold. The initial weights are independently drawn such that \( w_r \sim \mathcal{N}(0, I) \) and \( a_r \sim \text{Unif}(\{-1, 1\}^d) \) for all \( r \in [m] \). If \( m \geq C\frac{n_d^2 \lambda \delta}{\lambda_0^2} \) for some large enough constant \( C \), then with probability at least 1 − \( \delta \),

\[
\|X - U(t)\|_F^2 \leq \exp\left(-\frac{\lambda_0 t}{d}\right)\|X - U(0)\|_F^2.
\]
Remark 2: We initialize the second-layer weights $A$ with independent Rademacher entries. This is for convenience of analysis because such $A$ has constant-norm columns. However, similar results should easily follow for initialization with more practical schemes (for example, i.i.d. Gaussians).

By appropriately discretizing the gradient flow, we obtain a convergence result for gradient descent with finite step size for the jointly-trained regime:

**Theorem 7**: Suppose Assumptions 1 and 2 hold. At initialization, suppose the weights are independently drawn from $w_r \sim N(0, I)$ and $a_r \sim \text{Unif}((-1, 1)^d)$ for all $r \in [m]$. If $m \geq C\frac{\lambda^2}{\Delta^2}$ for some large enough constant $C$, then with probability at least $1 - \delta$ the gradient descent on $W$ with step size $\eta = \Theta(\frac{\lambda}{\sqrt{n}\lambda^2})$,

$$\|X - U(k)\|^2_F \leq (1 - \frac{\eta\lambda_0}{2d})^k\|X - U(0)\|^2_F.$$

Theorem 7 is our main contribution in terms of proving the linear convergence of jointly-trained autoencoders. In this regime, we obtain a significantly improved bound over Theorem 5 and similarly [4, Theorem 3.3]. Our result is consistent with [5, Theorem 6.3], but we have both layers jointly trained. The proof in [5, Theorem 6.3] which bounds Jacobian perturbations does not seem to be easily extended to the jointly trained case. The proofs of Theorem 6 and Theorem 7 are provided in Appendix D.

**VIII. WEIGHT-TIED AUTOENCODERS**

We conclude with the case of training two-layer autoencoders whose weights are shared (i.e., $A = W$). This is a common architectural choice in practice, and indeed previous theoretical analysis for autoencoders [14], [15], [23] have focused on this setting. We will show that somehow surprisingly, allowing the network to be over-parameterized in this setting leads to certain degeneracies. First, we prove:

**Lemma 2**: Let $x$ be any fixed sample. The weight $W$ is randomly initialized such that $w_r \sim N(0, \sigma^2 I)$ independently for $r = 1, 2, \ldots, m$, then

$$\mathbb{E}_{w_r \sim N(0, \sigma^2 I), v_r} [\|x - \frac{1}{m} W \phi(W^T x)\|^2] = \left(\frac{\sigma^2}{2} - 1\right)^2 \|x\|^2 + \frac{2(d + 3)\|\sigma^2 x\|^2}{4m}.$$

Particularly, when $\|x\| = 1, \sigma^2 = 2$, then

$$\mathbb{E}_{w_r \sim N(0, 2^2 I), v_r} [\|x - \frac{1}{m} W \phi(W^T x)\|^2] = \frac{2d + 3}{m}.$$

For an arbitrary small $\epsilon > 0$, the expected reconstruction loss is at most $\epsilon$ if $m \geq \Omega(d/\epsilon)$.

**Remark 3**: This Lemma has a few interesting implications. First, when $\sigma^2 = 2$, then

$$\mathbb{E}_{w_r \sim N(0, 2I), v_r} [\|x - u\|^2] = \frac{(2d + 3)\|x\|^2}{m},$$

which does not exceed $\epsilon$ if $m \geq 3d/\epsilon$ for $\epsilon > 0$. Provided that the data samples are normalized, if $m$ is sufficiently large, even with random initialization the reconstruction loss is very close to zero without any need for training. Therefore, mere over-parameterization already gets us to near-zero loss; the autoencoder mapping $\frac{1}{m} W \phi(W^T x) \approx x$ for any unit-norm $x$. It suggests that training of weight-tied autoencoders under high levels of over-parameterization may be degenerated.

The proof of Lemma 2 is given in Appendix E.

**APPENDIX A**

**USEFUL FACTS**

**Lemma 3** (Stein’s Lemma, e.g., [23]): For a random vector $x \in R^{d}$ such that $w \sim N(0, I)$ and function $h(w) : R^{d} \rightarrow R^{k}$ is weakly differentiable with Jacobian $D_{wh}$, we have

$$\mathbb{E}_{w \sim N(0, I)} [wh(w)^T] = \mathbb{E}_{w \sim N(0, I)} [(D_{wh})^T].$$

**Lemma 4**: Denote $S_{i} = \{r \in [m] : I[w_r(k + 1)^T x_i \geq 0] = \mathbb{I}[w_r(k)^T x_i \geq 0]\}$. If $\|w_r(k) - w_r(0)\| \leq R$, then

$$\sum_{r=1}^{m} \mathbb{I}[r \in S_{i}^{c}] \leq 4mR$$

with probability at least $1 - n \exp(-mR)$. This result is borrowed from the proof of [24, Claim 4.10].

**APPENDIX B**

**NEURAL TANGENT KERNEL**

**Proof of Lemma 1**: We prove this using simple calculus. The gradient of the loss over the parameters $\theta$ is

$$\nabla_{\theta} L(\theta) = -\sum_{i=1}^{n} \frac{\partial u_i^T}{\partial \theta} (x_i - u_i),$$

where $\partial u_i/\partial \theta$ denotes the Jacobian matrix of the output vector $u_i$ with respect to $\theta$. Combining with (III.1), the continuous-time dynamics of the prediction for each sample $i \in [n]$ is specified as

$$\frac{du_i}{dt} = \frac{\partial u_i}{\partial \theta} (-\nabla_{\theta} L(\theta)) = \sum_{j=1}^{n} \frac{\partial u_i}{\partial \theta} \frac{\partial u_j^T}{\partial \theta} (x_j - u_j).$$

Vectorizing $\frac{dU(t)}{dt}$, we get

$$\frac{d\text{vec}(U(t))}{dt} = K(t) \text{vec}(X - U(t)),$$

where $K(t)$ (or $K$) is an $nd \times nd$ matrix whose $(i,j)$-block is of size $d \times d$:

$$K_{i,j} = \frac{\partial u_i}{\partial \theta} \frac{\partial u_j^T}{\partial \theta} = \left(\frac{\partial}{\partial \theta} f(\theta, x_i)\right) \cdot \left(\frac{\partial}{\partial \theta} f(\theta, x_j)\right)^T.$$

One can easily verify that $K(t)$ is positive semi-definite. Note that in the supervised learning setting with a single output, the $(i,j)$-th block is a single scalar equal to the inner product of two gradients. \hfill \blacksquare
APPENDIX C

PROOFS OF SECTION VI: WEAK TRAINING

A. Proof Sketch

Our proof proceeds with three steps:

(i) We explicitly derive the continuous flow of the reconstructions $U(t) = [u_1(t), u_2(t), \ldots, u_n(t)]$ in C.4. The continuous flow can be viewed as the execution of gradient descent with infinitesimal learning rate.

(ii) To derive explicit convergence bounds, we will first show in Lemma 5 that $\lambda_{\min}(K_1(0)) \geq 3\lambda_0/4$ with high probability, using crude concentration arguments over the random initialization. Then, we will upper-bound $\|K_1(t) - K_1(0)\| \leq 2n^2d\delta/\sigma$ — the movement of each individual weight vector from the initial guess in terms of spectral norm. This allows us to achieve the bound in Theorem 5.

(iii) We discretize the continuous-time gradient flow and obtain analogous bounds for gradient descent in Section C-C.

B. Gradient Flow

Let us derive the neural tangent kernel for this training regime. We first calculate the gradient of $L(W)$ with respect to $W$. Since $A\phi(W^TX) = \sum_{r=1}^m a_r \phi(w_r^TX)$ for any $x \in \mathbb{R}^d$, it is convenient to compute the gradient with respect to each column $w_r$. The gradient $\nabla_{w_r} L(W)$ of the loss over $w_r$ is given by:

$$\nabla_{w_r} L = -\frac{1}{\sqrt{md}} \sum_{i=1}^n J_r(u_i)^\top (x_i - u_i)$$

$$= -\frac{1}{\sqrt{md}} \sum_{i=1}^n \mathbb{I}[w_r^TX_i \geq 0]x_i a_r a_r^\top (x_i - u_i),$$

(C.1)

where $J_r(u_i)^\top$ denotes the Jacobian matrix of the output vector $u_i$ with respect to $w_r$:

$$J_r(u_i) = \frac{1}{\sqrt{md}} a_r x_i^\top \phi'(w_r^TX_i) = \frac{1}{\sqrt{md}} \mathbb{I}[w_r^TX_i \geq 0]a_r x_i^\top.$$  

(C.2)

The gradient flow for the weight vector $w_r(t)$ is given by the following ODE:

$$\frac{d}{dt}w_r(t) = -\nabla_{w_r} L(W(t)).$$

(C.3)

Using (C.1) and (C.3), the continuous-time dynamics of the prediction for each sample $i \in [n]$ is:

$$\frac{d}{dt}u_i = \sum_{j=1}^n \left( \sum_{r=1}^m J_r(u_i)J_r^\top(u_j) \right) (x_j - u_j).$$

Vectorizing $\frac{dU(t)}{dt}$, we get the equation that characterizes the dynamics of $U(t)$:

$$\frac{d\text{vec}(U(t))}{dt} = \frac{1}{d} K_1(t) \text{vec}(X - U(t)),$$

(C.4)

where $K_1(t)$ is the $nd \times nd$ matrix whose $(i,j)$-block is of size $d \times d$ and defined as

$$K_1(t)_{i,j} = \frac{1}{m} \sum_{r=1}^m \mathbb{I}[w_r^T(t) X_{i,j} \geq 0] x_i^\top x_j a_r a_r^\top.$$
By the above argument, \( \|E_w[(X_t^T \tilde{X}_r)]^2\| \leq \lambda_n^2 \), so \( \sum_{w} E((Z_r - \tilde{Z}_r)^2) \leq m d \lambda_n^2 \).

From matrix Bernstein’s inequality [25, Theorem 1.4 of],
\[
P\left[ \|m K_1(0) - m K_1^\infty\| \geq \epsilon \right] \leq nd \exp \left( \frac{-\epsilon^2/2}{(d + 1) \lambda_n / 3 + m d \lambda_n^2} \right).
\]

Since the second term in the denominator of the exponent dominates \( \lambda_0 \leq \lambda_n \), we get
\[
m \geq C \frac{\lambda_n^2 d \log(\alpha/d)}{\lambda_n^2}
\]
where we pick \( \epsilon = m \lambda_0 / 4. \) Therefore,
\[
\|K_1(0) - K_1^\infty\| \leq \lambda_0 / 4
\]
with probability at least \( 1 - \delta \) for any \( \delta \in (0, 1) \). By Weyl’s inequality, we have with the same probability:
\[
\lambda_{\min}(K_1(0)) \geq 3 \lambda_0 / 4.
\]

The next step in our analysis is to upper bound the spectral norm of the kernel perturbation, \( \|K_1(t) - K_1(0)\| \), with high probability.

**Lemma 6:** Suppose \( w_r \sim N(0, I) \) and \( a_r \sim \text{Unif}\{\pm 1\}^d \) are drawn independently for all \( r \in [m] \). For any \( \delta \in (0, 1) \) and some \( R > 0 \), with probability at least \( 1 - \delta \):
\[
\sup_{\tilde{w} = (\tilde{w}_1, \ldots, \tilde{w}_m)} \left\{ \|K_1(\tilde{w}) - K_1(w)\| \leq \frac{2n^2 d R}{\delta} \right\},
\]
where \( K_1(w) = \frac{1}{m} \sum_{r=1}^m X(w_r)^T X(w_r) \otimes a_r a_r^T \).

**Remark 5:** One may ask why not to directly bound \( K_1(t) - K_1(0) \) for each time \( t \) but need the supremum over the ball near each \( w_r \). Basically, since \( w(t) \) depends on \( W(0) \) and \( A(0) \), directly working on \( K_1(t) - K_1(0) \) is difficult. The uniform bound (C.6) allows us to overcome this dependence when applied to \( K_1(t) - K_1(0) \).

Note that in this lemma we use \( K_1(w) \) to indicate that the kernel \( K \) is being evaluated at the weight vectors \( w_r \) and ignore the time index \( t \). In this Lemma, we use \( X(w_r) \) to denote \( X_r \) evaluated at \( w_r \).

**Proof:** For simplicity of notation, we use \( \sup_{\tilde{w}} \) to represent the supremum in (C.6), and \( \sup_{w_r} \) to represent \( \sup_{\|w_r - \tilde{w}_r\| \leq R} \). To prove this lemma, we work on the Frobenius norm instead of the spectral norm. Let us first write
\[
z_{ijr} = 1[w_r^T x_i \geq 0, w_r^T x_j \geq 0] - 1[w_r^T x_i \geq 0, w_r^T x_j \geq 0].
\]
Next,
\[
\|K_1(\tilde{w}) - K_1(w)\| \leq \frac{1}{m^2} \sum_{i,j=1}^m \sum_{r=1}^m z_{ijr} a_r a_r^T \|F_r\|_F
\]
\[
\leq \frac{1}{m^2} \left( \sum_{i,j=1}^m \sum_{r=1}^m z_{ijr} a_r a_r^T \|F_r\|_F \right)^2.
\]

The last step follows from the fact that \( |x_i^T x_j| \leq 1 \) due to Cauchy-Schwartz. Therefore,
\[
\sup_{\tilde{w}_r} \|K_1(\tilde{w}) - K_1(w)\| \leq \frac{1}{m^2} \sum_{i,j=1}^m \sum_{r=1}^m z_{ijr} a_r a_r^T \|F_r\|_F
\]
\[
\leq \frac{1}{m^2} \sum_{i,j=1}^m \sum_{r=1}^m z_{ijr} \|a_r a_r^T \|_F
\]
\[
\leq \frac{d}{m} \sum_{r=1}^m \sum_{i,j=1}^m z_{ijr},
\]
since \( \|a_r a_r^T \|_F = \|a_r\|^2 = d \). Now we take expectation over the random vector \( w_r \)’s on both sides:
\[
E_w[\sup_{\tilde{w}_r} \|K_1(\tilde{w}) - K_1(w)\|] \leq \frac{d}{m} \sum_{i,j=1}^m E_w[\sup_{\tilde{w}_r} z_{ijr}].
\]

Next, we bound \( E_w[\sup_{\tilde{w}_r} z_{ijr}] \). By definition of \( z_{ijr} \),
\[
|z_{ijr}| = |1[w_r^T x_i \geq 0, w_r^T x_j \geq 0] - 1[w_r^T x_i \geq 0, w_r^T x_j \geq 0]|
\]
\[
\leq |1[w_r^T x_i \geq 0] - 1[w_r^T x_i \geq 0]|
\]
\[
+ |1[w_r^T x_j \geq 0] - 1[w_r^T x_j \geq 0]|
\]
\[
\leq 1[w_r^T x_i \leq R] + 1[w_r^T x_j \leq R].
\]

The last step follows from the results in [4, Lemma 3.2]. So we get
\[
E_w[\sup_{\tilde{w}_r} z_{ijr}] \leq E_w[1[w_r^T x_i \leq R]] + 1[w_r^T x_j \leq R]]
\]
\[
= 2P_z \sim N(0, 1) \|z\| < R \leq \frac{4R}{\sqrt{2\pi}} < 2R.
\]

Therefore,
\[
E_w[\sup_{\tilde{w}_r} \|K_1(\tilde{w}) - K_1(w)\|] \leq \frac{2n^2 d R}{\delta}.
\]

Finally, by Markov’s inequality, with probability at least \( 1 - \delta \):
\[
\sup_{\tilde{w}_r} \|K_1(\tilde{w}) - K_1(w)\| \leq \frac{2n^2 d R}{\delta}.
\]

**Corollary 1:** Suppose \( \|w_r(t) - w_r(0)\| \leq R = \frac{\lambda_0 d}{\sqrt{m^2 d}} \) for all \( r \in [m] \) and \( t \geq 0 \) with probability at least \( 1 - \delta \). We have
\[
\lambda_{\min}(K_1(t)) > \frac{\lambda_0}{2}
\]
with probability at least \( 1 - 3\delta \) if \( m \geq C \lambda_0^2 d \log(\alpha/d) \).

**Proof:** This is the direct consequence of Lemma 5 and Lemma 6. Since \( \|w_r(t) - w_r(0)\| \leq R = \frac{\lambda_0 d}{\sqrt{m^2 d}} \) with probability at least \( 1 - \delta \) for all \( t \geq 0 \), then
\[
\|K_1(t) - K_1(0)\| \leq 2n^2 d R \delta = \frac{\lambda_0}{4}
\]
with probability at least \( 1 - 3\delta \). Using Weyl’s inequality, we can bound:
\[
\lambda_{\min}(K_1(t)) \geq \lambda_{\min}(K_1(0)) - \|K_1(t) - K_1(0)\| > \frac{\lambda_0}{2}
\]
with probability at least \( 1 - 3\delta \) if \( m \geq C \lambda_0^2 d \log(\alpha/d) \) as stated in Lemma 5.
In what follows, we show that $\|w_r(t) - w_r(0)\| \leq R$ with high probability if $m$ is sufficiently large.

**Lemma 7:** Fix $t > 0$. Suppose $\lambda_{\min}(K_1(s)) \geq \lambda_0/2$ for all $0 \leq s < t$. Then,
\[
\|X - U(s)\|^2_F \leq \exp\left(-\frac{\lambda_0 s}{d}\right) \|X - U(0)\|^2_F.
\]
Also, for each $r = 1, 2, \ldots, m$:
\[
\|w_r(t) - w_r(0)\| \leq \frac{d\sqrt{\lambda_n}\|X - U(0)\|_F}{\sqrt{m\lambda_0}} \triangleq R'.
\]

**Proof:** For all $s \in [0, t)$, we have
\[
\frac{d}{ds}\|\text{vec}(X - U(s))\|^2_F = -2\text{vec}(X - U(s))^T \frac{1}{d} K_1(s) \text{vec}(X - U(s)) \\
\leq -\frac{2}{d} \lambda_{\min}(K_1(s)) \|\text{vec}(X - U(s))\|^2_F \\
\leq -\frac{\lambda_0}{d} \|X - U(s)\|^2_F.
\]
by the assumption $\lambda_{\min}(K_1(s)) \geq \lambda_0/2$. Therefore, the loss at time $s$ is upper-bounded by
\[
\|X - U(s)\|^2_F = \|\text{vec}(X - U(s))\|^2_F \\
\leq \exp\left(-\frac{\lambda_0 s}{d}\right) \|\text{vec}(X - U(0))\|^2_F \\
\leq \exp\left(-\frac{\lambda_0 s}{d}\right) \|X - U(0)\|^2_F
\]
which decays exponentially with time $s$ at rate $\lambda_0/d$.

To upper bound the movement of the weights $\|w_r(t) - w_r(0)\|$, we use the above result while expanding the derivative of $w_r(s)$ over time $0 \leq s < t$:
\[
\frac{d}{ds}w_r(s) = \frac{-1}{\sqrt{md}} \sum_{i=1}^{n} \mathbb{1}_{[w_r^T x_i \geq 0]} x_i a_r^T (x_i - u_i(s)) \\
= \frac{1}{\sqrt{md}} \sum_{i=1}^{n} x_i a_r^T (x_i - u_i(s)) \\
\leq \frac{1}{\sqrt{md}} \sum_{i=1}^{n} \|x_i\|\|a_r\| \|X - U(s)\|_F \\
\leq \frac{\lambda_n}{m} \lambda_n \exp\left(-\frac{\lambda_0 s/d}{d}\right) \|X - U(0)\|_F,
\]
where the last step follows from $\|a_r\|^2 = d_i$ and $\|X\|^2 = \lambda_n$ and Eq. (C.8) from the differential equation, $w_r(s)$ is continuous for all $s \in [0, t)$, and so is $\|w_r(s) - w_r(0)\|$. Consequently, we can take the limit for $t' \to t$:
\[
\|w_r(t) - w_r(0)\|_2 = \lim_{t' \to t} \|w_r(t') - w_r(0)\|_2 \\
\leq \lim_{t' \to t} \int_{0}^{t'} \left[\frac{d}{ds}w_r(s)\right] ds \\
\leq \lim_{t' \to t} \int_{0}^{t'} \frac{d\sqrt{\lambda_n}\|X - U(0)\|_F}{\sqrt{m\lambda_0}} \exp\left(-\frac{\lambda_0 s/d}{d}\right) \|X - U(0)\|_F ds \\
\leq \frac{d\sqrt{\lambda_n}\|X - U(0)\|_F}{\sqrt{m\lambda_0}} \triangleq R',
\]

since $\exp(-\lambda_0 s/d)$ is continuous at $s = t$. Therefore, we finish the proof.

**Lemma 8:** If $R' < R$, then $\lambda_{\min}(K_1(t)) \geq \frac{1}{2}\lambda_0$ for all $t \geq 0$. Moreover, $\|w_r(t) - w_r(0)\| \leq R'$ and $\|X - U(t)\|^2_F \leq \exp(-\frac{\lambda_0 t}{d})\|X - U(0)\|^2_F$ for all $r \in [m]$.

**Proof:** We will prove this by contradiction. Assume the conclusion does not hold, meaning there exists $t_0$ such that:
\[
t_0 = \inf\{ t > 0 : \lambda_{\min}(H(t)) \leq \lambda_0/2 \}.
\]
We will argue that $t_0 > 0$ using the continuity. Since $w_r(t)$ is continuous in $t$, $K_1(t)$ and $\lambda_{\min}(K_1(t))$ are also continuous. Therefore, there exists $t' > 0$ such that for any $0 < \epsilon < \lambda_0/4$ we have
\[
\lambda_{\min}(K_1(t)) \geq \lambda_{\min}(K_1(0)) - \epsilon > \lambda_0/2.
\]
Since $t_0 > 0$, then for any $0 \leq s < t_0$, $\lambda_{\min}(H(s)) \geq \lambda_0/2$. By Lemma 7, we have for all $r \in [m]$:
\[
\|w_r(t_0) - w_r(0)\| \leq R' < R.
\]
Corollary 1 implies that $\lambda_0(H(t_0)) > \lambda_0/2$, which is a contradiction.

Therefore, we have proved the first part. For the second part, we have for all $t \geq 0$, $\lambda_{\min}(K_1(t)) \geq \frac{1}{2}\lambda_0$ and it follows from Lemma 7 that: $\|w_r(t) - w_r(0)\| \leq R'$ for all $r \in [m]$ and $\|X - U(t)\|^2_F \leq \exp(-\frac{\lambda_0 t}{d})\|X - U(0)\|^2_F$.

Now, we bound $\|X - U(0)\|^2_F$ to upper bound $R'$.

**Claim 1:** For any $\delta \in (0, 1)$, then $\|X - U(0)\|^2_F \leq \frac{32n}{\delta}$ with probability at least $1 - \delta$.

**Proof:** We prove this using Markov’s inequality. We use the independence between $A(0)$ and $W_0$ to derive expressions for the expectation. In this proof, the expectations are evaluated over $W(0)$ and $A(0)$.
\[
\mathbb{E}[\|X - U(0)\|^2_F] = \|X\|^2_F + \frac{1}{md}\mathbb{E}[\|A(0)\|^2 \phi(W(0)^T X)] \\
= n + \frac{1}{md}\mathbb{E}[\text{tr}(\phi(X^T W(0)) A(0) \phi(W(0)^T X)] \\
= n + \frac{1}{md}\mathbb{E}[\text{tr}(\phi(X^T W(0)) A(0) \phi(W(0)^T X))] \\
= n + \frac{1}{md}\mathbb{E}[\mathbb{E}[\phi(X^T W(0)) \phi(W(0)^T X)] \\
= n + \sum_{i=1}^{n} \mathbb{E}_w[\phi(x_i^T)^2] \\
= n + n\mathbb{E}_{x \in N(0, 1)}[z^2 \mathbb{1}[z \geq 0]] = \frac{3n}{2},
\]
where in the fourth step we use $\mathbb{E}[\phi(A(0)^T A(0))] = dI$, and in the last step we use the independence of the columns of $W(0)$. Using Markov, we get:
\[
\|X - U(0)\|^2_F \leq \frac{2n}{\delta}
\]
with probability at least $1 - \delta$.

**Proof of Theorem 4:** If the following condition holds
\[
R' = \frac{d\sqrt{\lambda_n}\|X - U(0)\|_F}{\sqrt{m\lambda_0}} \leq R = \frac{\delta\lambda_0}{8n^2d},
\]
then Lemma 7 follows. Using the condition with the bound $\|X - U(0)\|^2_F \leq \sqrt{2n/\delta}$ in Claim 1, we obtain $m = \frac{d\sqrt{\lambda_n}\|X - U(0)\|_F}{\sqrt{m\lambda_0}} \triangleq R'$.
\[ \Omega \left( n^{2d^2} k \right). \] This bound dominates the order of \( m \) required for the concentration of \( K_i(0) \) in the Corollary 1, and therefore Theorem 4 follows.

C. Gradient Descent

The above result for gradient flow can be viewed as a convergence rate for gradient descent in the weakly-trained regime with infinitesimally small step size. We restate and prove the convergence of gradient descent with finite step sizes.

**Theorem 8:** Suppose Assumptions 1 and 2 hold. The initial weights are independently drawn such that \( w_r \sim \mathcal{N}(0, I) \) and \( a_r \sim \text{Unif}(\{\pm 1\}^d) \) for all \( r \in [m] \). If \( m \geq C n^{2d^2} k \) for some large enough constant \( C \), then with probability at least \( 1 - \delta \) the gradient descent on \( W \) with step size \( \eta = \Theta \left( \frac{1}{\sqrt{m} \lambda_0} \right) \) in (C.1)

\[
\|X - U(k)\|_F^2 \leq \left( 1 - \frac{\eta \lambda_0}{2d} \right)^k \|X - U(0)\|_F^2 \tag{C.9}
\]

for \( k = 0, 1, \ldots \).

We will prove Theorem 5 by induction. The base case when \( k = 0 \) is trivially true. Assume Eq. (C.9) holds for \( k' = 0, 1, \ldots, k \); then we show it holds for \( k' = k + 1 \). To this end, first we prove \( \|w_r(k+1) - w_r(0)\| \) is small enough; then we use that property to bound \( \|X - U(k+1)\|_F^2 \).

**Lemma 9:** If (C.9) holds for \( k = 0, 1, \ldots, k \), then we have for all \( r \in [m] \),

\[
\|w_r(k+1) - w_r(0)\| \leq \frac{4d \sqrt{n} \|X - U(0)\|_F}{\sqrt{m} \lambda_0} \triangleq R'.
\]

**Proof:** We use the expression of the gradient in (C.1), which is:

\[
\nabla_{w_r} L(W(k)) = -\sum_{i=1}^n \frac{1}{\sqrt{md}} \mathbb{1}[w_r(k)^T x_i \geq 0] x_i a_r^T (x_i - u_i(k))
\]

\[
= -\frac{1}{\sqrt{md}} \tilde{X}_r(k)(X - U(k))^T a_r.
\]

Then, the difference of the weight vector \( w_r \) is:

\[
\|w_r(k+1) - w_r(0)\| = \eta \left\| \sum_{k'=0}^k \nabla_{w_r} L(w_r(k')) \right\| = \eta \left\| \sum_{k'=0}^k \frac{1}{\sqrt{md}} \tilde{X}_r(k')(X - U(k'))^T a_r \right\|
\]

\[
\leq \eta \frac{\|X\|}{\sqrt{md}} \sum_{k'=0}^k \|\text{vec}(X - U(k'))\|_F \|a_r\|,
\]

\[
\leq \eta \frac{\sqrt{n}}{\sqrt{md}} \sum_{k'=0}^k \left( 1 - \frac{\eta \lambda_0}{2} \right)^{k'/2} \|X - U(0)\|_F
\]

\[
\leq \eta \frac{\sqrt{n}}{\sqrt{md}} \|X - U(0)\|_F \sum_{k'=0}^k \left( 1 - \frac{\eta \lambda_0}{2d} \right)^{k'/2}
\]

\[
= \eta \frac{\sqrt{n}}{\sqrt{md}} \|X - U(0)\|_F \frac{1}{\eta \lambda_0 / (4d)}
\]

\[
= \frac{4d \sqrt{n} \|X - U(0)\|_F}{\sqrt{m} \lambda_0},
\]

where the third step and the fourth step follow from the facts that \( \|\tilde{X}_r(k')\| \leq \|X\| = \sqrt{n} \) and \( \|a_r\| = \sqrt{d} \). The last step follows because \( \sum_{i=0}^\infty (1 - \eta \lambda_0 / 2d)^{i/2} \leq \frac{4d}{\eta \lambda_0}. \)

Now, let us derive the form of \( X - U(k+1) \). First, we compute the difference of the prediction between two consecutive steps, similar to deriving \( \frac{du_i}{dt} \). For each \( i \in [n] \), we have

\[
u_i(k+1) - u_i(k) = \frac{1}{\sqrt{md}} \sum_{r=1}^m a_r (\phi(w_r(k+1)^T x_i) - \phi(w_r(k)^T x_i)). \tag{C.10}
\]

Substituting \( w_r(k+1) = w_r(k) - \eta \nabla w_r L(W(k)) \), we split the right hand side into two parts: \( v_{1,i} \) represents the terms that the activation pattern does not change and \( v_{2,i} \) represents the remaining term that pattern may change. Formally speaking, for each \( i \in [n] \), we define

\[ S_i = \{ r \in [m] : \mathbb{1}[w_r(k+1)^T x_i \geq 0] = \mathbb{1}[w_r(k)^T x_i \geq 0] \}, \]

and \( S_i^+ = [m] \setminus S_i \). Then, we can formally define \( v_{1,i} \) and \( v_{2,i} \) as follows:

\[ v_{1,i} \triangleq \frac{1}{\sqrt{md}} \sum_{r \in S_i} a_r (\phi(w_r(k+1)^T x_i) - \phi(w_r(k)^T x_i)), \]

\[ v_{2,i} \triangleq \frac{1}{\sqrt{md}} \sum_{r \in S_i^+} a_r (\phi(w_r(k+1)^T x_i) - \phi(w_r(k)^T x_i)). \]

We write \( v_1 = (v_{1,1}, v_{1,2}, \ldots, v_{1,n})^T \) and do the same for \( v_2 \), so

\[ \text{vec}(U(k+1) - U(k)) = v_1 + v_2. \]

In order to analyze \( v_1 \in \mathbb{R}^n \), we define \( K \) and \( K_i^+ \) in \( \mathbb{R}^{n \times d} \) as follows:

\[ K(k)_{i,j} = \frac{1}{m} \sum_{r=1}^m x_i^T x_j \mathbb{1}[w_r(k)^T x_i \geq 0, w_r(k)^T x_j \geq 0] a_r a_r^T, \]

\[ K_i(k)_{i,j} = \frac{1}{m} \sum_{r \in S_i} x_i^T x_j \mathbb{1}[w_r(k)^T x_i \geq 0, w_r(k)^T x_j \geq 0] a_r a_r^T. \]

Next, we write \( \phi(z) = z \mathbb{1}[z \geq 0] \) to make use of the definition of \( S_i \) and expand the form of \( \nabla_{w_r} L(W(k)) \):

\[ v_{1,i} = \frac{1}{\sqrt{md}} \sum_{r \in S_i} a_r \left( -\eta \nabla_{w_r} L(W(k)) \right)^T x_i \mathbb{1}[w_r(k)^T x_i \geq 0] \]

\[ = \frac{\eta}{md} \sum_{j=1}^n x_i^T x_j \sum_{r \in S_i} \mathbb{1}[w_r(k)^T x_i \geq 0, w_r(k)^T x_j \geq 0] a_r a_r^T (x_j - u_j), \]

\[ = \frac{\eta}{d} \sum_{j=1}^n (K_i(k) - K_i^+(k))(x_j - u_j), \]

Then, we can write \( v_1 \) as:

\[ v_1 = \frac{\eta}{d} \left( K_1(k) - K_1^+(k) \right) \text{vec}(X - U(k)), \tag{C.11} \]
and expand $\|X - U(k + 1)\|^2_F$:

$$
\|X - U(k + 1)\|^2_F = \|\text{vec}(X - U(k + 1))\|^2 \\
= \|\text{vec}(X - U(k)) - \text{vec}(U(k + 1) - U(k))\|^2_F \\
= \|X - U(k)\|^2_F - 2\text{vec}(X - U(k))^\top \text{vec}(U(k + 1) - U(k)) + \|U(k + 1) - U(k)\|^2_F.
$$

We can further expand the second term above using (D.14) as below:

$$
\text{vec}(X - U(k))^\top \text{vec}(U(k + 1) - U(k)) \\
= \text{vec}(X - U(k))^\top (v_1 + v_2) \\
= \text{vec}(X - U(k))^\top v_1 + \text{vec}(X - U(k))^\top v_2 \\
= \frac{\eta}{d} \text{vec}(X - U(k))^\top K_1(k) \text{vec}(X - U(k)) \\
- \frac{\eta}{d} \text{vec}(X - U(k))^\top K_1(k)^\perp (X - U(k)) + \text{vec}(X - U(k))^\top v_2.
$$

We define and bound the following quantities and bound them in Claims 2, 3, 4 and 5.

$$
C_1 = \frac{-2\eta}{d} \text{vec}(X - U(k))^\top K_1(k) \text{vec}(X - U(k)), \\
C_2 = \frac{2\eta}{d} \text{vec}(X - U(k))^\top K_1(k)^\perp (X - U(k)), \\
C_3 = -2\text{vec}(X - U(k))^\top v_2, \\
C_4 = \|U(k + 1) - U(k)\|^2_F.
$$

**Proof of Theorem 5:** We are now ready to prove the induction hypothesis. What we need to is to prove

$$
\|X - U(k')\|^2_F \leq \left(1 - \frac{\eta \lambda_0}{2d}\right) \|X - U(0)\|^2_F
$$

holds for $k' = k + 1$ with probability at least $1 - \delta$. In fact,

$$
\|X - U(k + 1)\|^2_F = \|X - U(k)\|^2_F + C_1 + C_2 + C_3 + C_4 \\
\leq \|X - U(k)\|^2_F \left(1 - \frac{\eta \lambda_0}{d} + 8\eta m R + 8\eta m R + \eta^2 n \lambda_n\right),
$$

with probability at least $1 - \delta$ where the last step follows from Claim 2, 3, 4, and 5.

1) **Choice of $\eta$ and $R$:** We need to choose $\eta$ and $R$ such that

$$
\left(1 - \frac{\eta \lambda_0}{d} + 8\eta m R + 8\eta m R + \eta^2 n \lambda_n\right) \leq 1 - \frac{\eta \lambda_0}{2d}.
$$

If we set $\eta = \frac{\lambda_n}{4nd\lambda_n}$ and $R = \frac{\lambda_0}{64n d}$, we have

$$
8\eta m R + 8\eta m R = 16\eta m R \leq \frac{\eta \lambda_0}{4d}, \quad \text{and} \quad \eta^2 n \lambda_n \leq \frac{\eta \lambda_0}{4d}.
$$

Finally,

$$
\|X - U(k + 1)\|^2_F \leq \left(1 - \frac{\eta \lambda_0}{2d}\right) \|X - U(k)\|^2_F.
$$

holds with probability at least $1 - \delta$ if $2n \exp(-m R) \leq \delta/3$.

2) **Lower Bound on the Level of Over-Parameterization $m$:** We require for any $\delta \in (0, 1)$ that

$$
R' = \frac{4d \sqrt{\lambda_n} \|X - U(0)\|_F}{\sqrt{m \lambda_0}} < R = \min \left\{ \frac{\lambda_0}{64nd}, \frac{\lambda_0 \delta}{2n^2 d} \right\},
$$

where the first bound on $R$ comes from the gradient descent whereas the second is required in Lemma 6. By Claim 1 that $\|X - U(0)\|_F \leq \sqrt{\frac{2n}{\delta}}$ with probability at least $1 - \delta$, then we require

$$
m \geq C \frac{\eta^5 \lambda_n d^4}{\lambda_0^5 \delta^4},
$$

for a sufficiently large constant $C > 0$ so that the descent holds with probability $1 - \delta$.

We give proofs for Claims 2, 3, 4, and 5 in the following:

**D. Proof of Supporting Claims**

To prove the bounds in Claims 2, 3, 4, and 5, we use the bound $\|w_r(k + 1) - w_r(0)\| \leq R'$ for all $r \in [m]$ in Claim 9. In what follows, we assume $R' < R$, which is the weight movement allowed to achieve Lemma 6. This assumption holds with high probability as long as $m$ is large enough.

**Claim 2:** Let $C_1 = -\frac{2d}{d} \text{vec}(X - U(k))^\top K_1(k) \text{vec}(X - U(k))$. Then we have

$$
C_1 \leq -\frac{\eta \lambda_0}{d} \|X - U(k)\|^2_F.
$$

with probability at least $1 - \delta$.

**Proof:** Using Lemma 9, we have $\|w_r(k) - w_r(0)\| \leq R' < R$ for all $r \in [m]$. By Lemma 6, we have

$$
\|K_1(k) - K_1(0)\| < \frac{\lambda_0}{4}.
$$

Therefore, $\lambda_{\min}(K_1(k)) \geq \frac{\lambda_0}{2}$ with probability at least $1 - \delta$.

As a result,

$$
\text{vec}(X - U(k))^\top K_1(k) \text{vec}(X - U(k)) \geq \frac{\lambda_0}{2} \|X - U(k)\|^2 = \frac{\lambda_0}{2} \|X - U(k)\|^2_F,
$$

and $C_1 \leq -\frac{\eta \lambda_0}{d} \|X - U(k)\|^2_F$ with probability at least $1 - \delta$. **Claim 3:** Let $C_2 = \frac{2d}{d} \text{vec}(X - U(k))^\top K_1(k)^\perp (X - U(k))$. We have

$$
C_2 \leq 8\eta m R \|X - U(k)\|^2_F.
$$

with probability at least $1 - n \exp(-m R)$.
Proof: All we need is to bound $K_1(k)^\perp$. A simple upper bound is

$$\|K_1(k)^\perp\|^2 \leq \sum_{i,j=1}^n \|K_1(k)^\perp_{i,j}\|^2_F$$

$$\leq \sum_{i,j=1}^n \left( \frac{1}{m} \sum_{r \in S_i^+} x_i^T x_j \mathbb{1}[w_r(k)^\top x_i \geq 0, w_r(k)^\top x_j \geq 0] a_r a_r^\top \right)$$

$$\leq d^2 \sum_{i,j=1}^n \left( \frac{1}{m} \sum_{r \in S_i^+} \mathbb{1}[r \in S_i^+] \right)^2$$

$$\leq 16n^2 d^2 R^2$$

with probability $1 - n \exp(-mR)$ where the last step follows from Lemma 4. Then, with that same probability $\|K_1(k)^\perp\|_F \leq 4ndR$, and

$$C_2 = \frac{2n}{d} \|\text{vec}(X - U(k))^\top K_1(k)^\perp (X - U(k))\|_F$$

$$\leq \frac{2n}{d} \|K_1(k)^\perp\|_F \|X - U(k)\|_F$$

$$\leq 8\eta R \|X - U(k)\|_F^2.$$  

Claim 4: Let $C_3 = -2\text{vec}(X - U(k))^\top v_2$, then with probability at least $1 - n \exp(-mR)$

$$C_3 \leq 8\eta R \|X - U(k)\|_F^2.$$  

Proof: We have $C_3 \leq 2\|X - U(k)\|_F \|v_2\|$. Using the Lipschitz property of $\phi$, we have

$$\|v_2\|^2 \leq \sum_{i=1}^n \|v_{2,i}\|^2$$

$$\leq \sum_{i=1}^n \left( \frac{1}{md} \sum_{r \in S_i^+} a_r \left( \phi(w_r(k + 1)^\top x_i) - \phi(w_r(k)^\top x_i) \right) \right)^2$$

$$\leq \frac{2^2}{m} \sum_{i=1}^n \left( \sum_{r \in S_i^+} \left( \nabla_{w_r} L(W(k))^\top x_i \right) \right)^2$$

$$\leq \frac{2^2}{m} \max_r \left\| \nabla_{w_r} L(W(k))^\top \right\|^2 \sum_{i=1}^n \left( \sum_{r \in S_i^+} \mathbb{1}[r \in S_i^+] \right)^2$$

$$\leq \frac{2^2 \lambda n}{m^2} \|X - U(k)\|_F^2 \sum_{i=1}^n \left( \sum_{r \in S_i^+} \mathbb{1}[r \in S_i^+] \right)^2$$

$$\leq \frac{2^2 \lambda n}{m^2} \|X - U(k)\|_F^2 \sum_{i=1}^n (4mR)^2$$

$$\leq 16n^2 R^2 \eta^2 \|X - U(k)\|_F^2$$

$$\leq 16n^2 R^2 \eta^2 \|X - U(k)\|_F^2.$$  

with probability $1 - n \exp(-mR)$. The sixth step we use

$$\|\nabla_{w_r} L(W(k))\| = \left\| \frac{1}{\sqrt{md}} \bar{X}_r(k)(X - U(k))^\top a_r \right\|$$

$$\leq \frac{\lambda n}{\sqrt{m}} \|X - U(k)\|_F,$$

and the last step follows from from Lemma 4 that $\sum_{r=1}^m \mathbb{1}[r \in S_i^+] \leq 4mR$ with probability at least $1 - n \exp(-mR)$. Substitute the bound into $C_3$, then we finish the proof.  

Claim 5: Let $C_4 = \|U(k + 1) - U(k)\|_F^2$. Then we have

$$C_4 \leq \eta^2 n\lambda_n \|X - U(k)\|_F^2.$$  

Proof: Previously in Lemma 9, we proved that

$$\|\nabla_{w_r} L(W(k))\| = \left\| \frac{1}{\sqrt{md}} \bar{X}_r(k)(X - U(k))^\top a_r \right\|$$

$$\leq \frac{\lambda n}{\sqrt{m}} \|X - U(k)\|_F.$$  

Expand the form of $U(k + 1) - U(k)$ and use the Lipschitz of ReLU to get

$$C_4 = \sum_{i=1}^n \|u_i(k + 1) - u_i(k)\|^2$$

$$= \frac{1}{md} \sum_{i=1}^n \left( \sum_{r=1}^m a_r \left( \phi(w_r(k + 1)^\top x_i) - \phi(w_r(k)^\top x_i) \right) \right)^2$$

$$\leq \eta^2 \sum_{i=1}^n \frac{1}{m} \left( \sum_{r=1}^m \left\| \nabla_{w_r} L(W(k)) \right\| \right)^2$$

$$\leq \eta^2 \sum_{i=1}^n \frac{1}{m} \left( \sum_{r=1}^m \frac{\lambda n}{\sqrt{m}} \|X - U(k)\|_F \right)^2$$

$$= \eta^2 n\lambda_n \|X - U(k)\|_F^2.$$  

Therefore, we finish the proof.

APPENDIX D

PROOFS FOR SECTION VII: JOINT TRAINING

A. Proof Sketch

Similar to the weak training (Appendix C), our proof proceeds with three steps:

(i) We explicitly derive the continuous flow of the reconstructions $\bar{U}(t) = [u_1(t), u_2(t), \ldots, u_n(t)]$ in D.5. The continuous flow can be viewed as the execution of gradient descent with infinitesimal learning rate.

(ii) To derive explicit convergence bounds, we will show in Lemma 10 that $\lambda_{\text{min}}(K_2(0)) \geq \lambda_{\text{min}}(H(0)) \geq 3\alpha_0/4$ with high probability, using crude concentration arguments over the random initialization. Then, we will upper-bound $\|K_2(t) - K_2(0)\|$ — the movement of each individual weight vector from the initial guess in terms of spectral norm. This allows us to achieve the bound in Theorem 7.

(iii) We discretize the continuous-time gradient flow and obtain analogous bounds for gradient descent in Section D-C.
B. Gradient Flow

Similar to the previous case, we derive the gradients of \(L(W, A)\) with respect to the column \(w_r\) of \(W\) and \(a_r\) of \(A\). The gradient \(\nabla_w L(W, A)\) is the same in (C.1) in Section C-B whereas \(\nabla_a L(W, A)\) is standard:

\[
\nabla_w L(W, A) = -\frac{1}{\sqrt{md}}\sum_{i=1}^{n} 1[w_r^T x_i \geq 0 | x_a^T (x_i - u_i) ,
\]

\[\tag{D.1}\]

\[
\nabla_a L(W, A) = -\frac{1}{\sqrt{md}}\sum_{i=1}^{n} \phi(w_r^T x_i)(x_i - u_i).
\]

\[\tag{D.2}\]

Consider two ODEs, one for each weight vector over the continuous time \(t\):

\[
\frac{dw_r(t)}{dt} = -\nabla_w L(W(t), A(t)),
\]

\[\tag{D.3}\]

\[
\frac{da_r(t)}{dt} = -\nabla_a L(W(t), A(t)).
\]

\[\tag{D.4}\]

Using (D.1), (D.2), (D.3) and (D.4), the continuous-time dynamics of the predicted output, \(u_i(t)\), for sample \(x_i\) is given by:

\[
\frac{du_i(t)}{dt} = \frac{d}{dt} \left( \frac{1}{\sqrt{md}} \sum_{r=1}^{m} a_r \phi(w_r^T x_i) \right)
\]

\[
= \frac{1}{\sqrt{md}} \sum_{r=1}^{m} \left( J_{w_r}(a_r \phi(w_r^T x_i)) \frac{dw_r}{dt} + J_{a_r}(a_r \phi(w_r^T x_i)) \frac{da_r}{dt} \right)
\]

\[
= \frac{1}{\sqrt{md}} \sum_{r=1}^{m} \sum_{j=1}^{n} 1[w_r^T x_i \geq 0, w_r^T x_j \geq 0 | x_a^T x_j x_a a^T (x_j - u_j)
\]

\[
= \frac{1}{\sqrt{md}} \sum_{r=1}^{m} \sum_{j=1}^{n} \phi(w_r^T x_i) \phi(w_r^T x_j) I(x_j - u_j).
\]

In these expressions, we skip the dependence of the weight vectors on time \(t\) and simply write them as \(w_r\) and \(a_r\). Vectorizing \(\frac{d}{dt} U(t)\), we get to the key equation that characterizes the dynamics of \(U(t)\):

\[
\frac{d}{dt} (G(t) + H(t)) \text{vec}(X - U(t)).
\]

\[\tag{D.5}\]

In the above equation, \(G(t)\) is a size-\(nd \times nd\) matrix of the form:

\[
G(t) = \frac{1}{m} \sum_{r=1}^{m} \bar{X}_r(t)^T \bar{X}_r(t) \otimes a_r(t)a_r(t)^T,
\]

\[\tag{D.6}\]

where

\[
\bar{X}_r(t) = \left[ 1[w_r^T x_1 \geq 0 | x_1 , \ldots , 1[w_r^T x_n \geq 0 | x_n \right],
\]

while \(H(t)\) is a size-\(nd \times nd\) matrix:

\[
H(t) = \frac{1}{m} \sum_{r=1}^{m} \phi(X^T w_r(t)) \phi(w_r(t)^T X) \otimes I.
\]

\[\tag{D.7}\]

Since \(G(0)\) is positive semi-definite, in order to bound the minimum eigenvalue of \(K_2(0)\), all we need is to bound that of \(H(0)\). Importantly, we observe that the smoothness of the kernel \(H(t)\) is much better as a function of the deviation of the weights from the initialization. This allows the weights to change with a larger amount than merely using \(G(t)\), and enables us to significantly reduce the number of parameters required for the gradient to reach a global optimum.

\textbf{Lemma 10:} For any \(\delta \in (0, \frac{\lambda_0}{2d})\), if \(m \geq C \frac{n d \lambda_n \log^{4}(nd/\delta)}{\lambda_0^{2}}\) for some large enough constant \(C\), then with probability at least \(1 - 1/(2nd)^{2 \log md - \delta}\), one obtains \(|H(0) - H^{\infty}| \leq \lambda_0/4\) and \(\lambda_{\text{min}}(H(0)) \geq 3 \lambda_0/4\).

Recall that

\[
H(0) = \frac{1}{m} \sum_{r=1}^{m} \phi(X^T w_r(0)) \phi(w_r(0)^T X) \otimes I,
\]

\[\tag{D.8}\]

and \(H^{\infty} = E_{H(0), A(0)}[H(0)]\). Our goal is to show the concentration of \(\sum_{r=1}^{m} \phi(X^T w_r(0)) \phi(w_r(0)^T X)\). Let us use \(w_r\) to mean \(w_r(0)\) and denote the \(r\text{-th}\)-random matrix as

\[
Z_r = \phi(X^T w_r) \phi(w_r^T X).
\]

We use Lemma B.7 of [26] and verify the required conditions by the next results.

\textbf{Claim 6 (Condition I for \(H(0)\)):} The following are true:

(i). \(|Z_r| \leq \lambda_n |w_r|^2\).

(ii). \(E_{w_r} |w_r|^2 \leq 4d \sqrt{\log (2/\delta)} \geq 1 - \delta\) for any \(\delta \in (0, 1)\).

\textbf{Proof:} (i) We have

\[
|Z_r| = \|\phi(X^T w_r) \phi(w_r^T X)\| = \|\phi(X^T w_r)\|^2
\]

\[
\leq \|X^T X\| \|w_r\|^2 \leq \lambda_n \|w_r\|^2,
\]

which gives the first part (i). For the second part, we use the fact \(|w_r|^2\) is a chi-squared random variable with \(d\) degrees of freedom, and sub-exponential with sub-exponential norm \(2\sqrt{d}\), meaning that:

\[
E_{w_r} \left( |w_r|^2 - d \right) \geq c \leq 2 \exp(-\frac{c^2}{8d})
\]

For any \(\delta \in (0, 1)\) and \(c^2 = 9d \log (2/\delta) \geq 8d \log (2/\delta)\), we have

\[
|w_r|^2 - d \leq 3 \sqrt{d \log (2/\delta)}
\]

with probability at least \(1 - \delta\). Then, \(|w_r|^2 \leq d + 3 \sqrt{d \log (2/\delta)} \leq 4d \sqrt{\log (2/\delta)}\) with probability at least \(1 - \delta\).

\textbf{Claim 7 (Condition II for \(H(0)\)):} \(|E[Z_r, Z_r^T]| \leq 3n \lambda_n\).

\textbf{Proof:} We have

\[
Z_r Z_r^T = \phi(X^T w_r) \phi(w_r^T X) \otimes \phi(X^T w_r) \phi(w_r^T X)
\]

\[
\leq \|\phi(X^T w_r)\|^2 \|w_r\|^2 X^T X
\]

\[
\leq \sum_{i=1}^{n} \phi(w_r^T x_i)^2 \|w_r\|^2 X^T X.
\]

We need is to compute \(E_{w_r} \phi(w_r^T x_i)^2 \|w_r\|^2\), which is already done in (E.4), Section VIII. To be precise, we have

\[
E_{w_r} \phi(w_r^T x_i)^2 \|w_r\|^2 = \frac{d+2}{2} \leq d
\]

where \(x_i = x_i(1|w_r^T x_i \geq 0\) for each \(i \in n\). Then, we can write

\[
|E_{w_r} [Z_r, Z_r^T]| \leq nd \lambda_n.
\]

\textbf{Claim 8 (Condition IV for \(H(0)\)):}

\[
sup_{|b|=1} (E\left(\left|b^T Z_r b\right|^2\right))^1/2 \leq \sqrt{3} d \lambda_n.
\]
**Proof:** Recall \( Z_r = \phi(X^T w_r)\phi(w_r^T X) \), and for any unit-norm vector \( b \in \mathbb{R}^n \)

\[
(b^T Z_r b)^2 = \|b^T \phi(X^T w_r)\|_2^4 \leq \|\phi(X^T w_r)\|_2^4 \leq \lambda_r^2 \|w_r\|_2^4.
\]

Moreover, \( \|w_r\|^2 \) is a chi-squared random variable with \( d \) degree of freedom, so

\[
\mathbb{E}[\|w_r\|^2] = 3d^2.
\]

Therefore, \( \sup_{\|b\|=1} (\mathbb{E}[b^T Z_r b^2])^{1/2} \leq \sqrt{3d} \lambda_r. \)

**Proof of Lemma 10:** With the conditions fulfilled in Claims 6, 7 and 8, we can now apply [26, Lemma B.7] to show the concentration of \( K_2(0) \):

\[
\frac{1}{m} \sum_{r=1}^{m} Z_r - \mathbb{E}[Z_r] \parallel \leq \epsilon \mathbb{E}[\|Z_r\|]
\]

with probability \( 1 - 1/n^2t - n \delta \) for any \( t \geq 1, \epsilon \in (0,1) \) and \( \delta < \epsilon \mathbb{E}[\|Z_r\|] / (2\sqrt{3d}\lambda_n)^2 \).

For the target bound, we choose \( \epsilon \mathbb{E}[\|Z_r\|] = \lambda_0/4 \), \( t = \log(2n) \) and note that \( \lambda_0 \leq \mathbb{E}[\|Z_r\|] = \|K_2^\infty\| \leq \lambda_n \). Therefore, with probability \( 1 - 1/(2n)d^\log nd - m \delta \) for any \( \delta \in (0, \frac{\lambda_0}{12d^\log nd}) \), then

\[
\|H(0) - H^\infty\| \leq \frac{\lambda_0}{4}
\]

if \( m \) satisfies

\[
m \geq 18 \log^2(2n) nd \lambda_n + \lambda_n^2 + 4d \sqrt{\log(2/\delta)} \lambda_n \lambda_0/4,
\]

which means \( m = C \frac{nd \lambda_n \log^2 nd \log(1/\delta)}{\lambda_0} \).

**Lemma 11:** Suppose \( \|W(t) - W(0)\|_F \leq R_w \). Then,

\[
\|H(t) - H(0)\| \leq \frac{\lambda_0}{m} (\|W(0)\| + R_w) R_w.
\]

Particularly, if \( \frac{\lambda_0}{d} (\|W(0)\| + R_w) R_w \leq \frac{\lambda_0}{2} \), then \( \|H(t) - H(0)\| \leq \frac{\lambda_0}{2} \). Therefore, \( \lambda_{\text{min}}(K_2(t)) > \frac{\lambda_0}{2} \) if \( \lambda_{\text{min}}(H(0)) \geq \frac{\lambda_0}{4} \).

**Remark 6:** Let us compare with Lemma 6. Note that compared with that bound, \( O(n^2dR_w) \) on the kernel perturbation, here the spectral norm bound on \( H(t) - H(0) \) is significantly better in two ways:

(i) the bound scales with \( 1/\sqrt{m} \), which later determines the over-parameterization.
(ii) the movement is now characterized by the total \( \|W(t) - W(0)\|_F \). This is possible due to the smoothness of the ReLU activation, which is the reason why we focus on \( H(t) \) instead of \( G(t) \).

**Proof:** We apply the triangle inequality and use the Lipschitz property of the rectified linear unit to bound the difference. Recall that

\[
H(t) = \frac{1}{m} \sum_{r=1}^{m} \phi(X^T w_r(t)) \phi(w_r(t)^T X)
\]

\[
= \frac{1}{m} \phi(X^T W(t)) \phi(W(t)^T X).
\]

Then, we can upper bound the perturbation as follows:

\[
\|H(t) - H(0)\| \\
\leq \frac{1}{m} \|\phi(X^T W(t)) \phi(W(t)^T X) - \phi(X^T W(0)) \phi(W(0)^T X)\| \\
\leq \frac{1}{m} \|\phi(X^T W(t)) \phi(W(t)^T X) - \phi(W(0)^T X)\| \\
+ \frac{1}{m} \|\phi(X^T W(t)) - \phi(X^T W(0))\| \|\phi(W(0)^T X)\| \\
\leq \frac{1}{m} \|X\| \|\phi(X^T W(t)) - \phi(W(0))\| \|W(0)\|_F + \frac{1}{m} \|\phi(X^T W(t)) - \phi(X^T W(0))\| \|W(t) - W(0)\|_F \\
\leq \frac{\lambda_0}{m} \|W(0)\|_F + \frac{1}{m} \|W(t) - W(0)\|_F \\
\leq \frac{\lambda_0}{m} (\|W(0)\| + R_w) \lambda_n R_w.
\]

In the third step, we use the fact that the ReLU function is 1-Lipschitz and \( \|\phi(X^T W(t))\| \leq \|X\| \|W(t)\|_F \). The last step follows by \( \|W(t) - W(0)\|_F \leq R_w \).

Using the condition and Weyl’s inequality, one can easily show that \( \lambda_{\text{min}}(K_2(t)) > \lambda_{\text{min}}(H(t)) > \lambda_0/2 \).

We have proved that as long as the weight matrix \( W(t) \) do not change much over \( t \), the minimum eigenvalue of \( K_2(t) \) stays positive. Next, we show that this implies the exponential decay of the loss with iteration, and give a condition under which the weights do not change much.

**Lemma 12:** Fix \( t > 0 \). Suppose \( \lambda_{\text{min}}(K_2(t)) \geq \frac{\lambda_0}{2} \) for all \( 0 \leq s < t \). Then

\[
\|X - U(s)\|_F^2 \leq \exp\left(-\frac{\lambda_0}{d} \right) \|X - U(0)\|_F^2.
\]

**Proof:** We have \( \lambda_{\text{min}}(K_2(s)) \geq \frac{\lambda_0}{2} \), then

\[
\frac{d}{ds} \|\text{vec}(X - U(s))\|_F^2
\]

\[
= -2 \text{vec}(X - U(s))^\top K_2(t)^{-1} \text{vec}(X - U(s)) \leq \frac{-2}{d} \lambda_{\text{min}}(K_2(t)) \|\text{vec}(X - U(s))\|_F^2
\]

\[
\leq -\frac{\lambda_0}{d} \|X - U(s)\|_F^2,
\]

since \( \lambda_{\text{min}}(H(s)) \geq \frac{\lambda_0}{2} \). Therefore,

\[
\|X - U(s)\|_F^2 = \|\text{vec}(X - U(s))\|_F^2
\]

\[
\leq \exp\left(-\frac{\lambda_0}{d} \right) \|\text{vec}(X - U(0))\|_F^2
\]

\[
\leq \exp\left(-\frac{\lambda_0}{d} \right) \|X - U(0)\|_F^2.
\]

**Lemma 13:** Fix \( t > 0 \). Suppose \( \lambda_{\text{min}}(K_2(s)) \geq \frac{\lambda_0}{2} \) and \( \|A(s) - A(0)\|_F \leq R_a \) for all \( 0 \leq s < t \). For all \( r \in [m] \), we have \( \|W(t) - W(0)\|_F \leq R_w \) where

\[
R_w^2 \triangleq \frac{2 \sqrt{d^2 \lambda_n (\|A(0)\| + R_a) \|X - U(0)\|_F}}{\sqrt{m} \lambda_0}.
\]
Proof: For $s \in [0, t)$, we have
\[
\frac{d}{ds} w_r(s) = -\nabla_{w_r} L(W(s), A(s))
\]
\[
= \frac{1}{\sqrt{md}} \sum_{i=1}^{n} 1[w_i^T x_i \geq 0]x_i a_r(s)^T (x_i - u_i(s))
\]
\[
= \frac{1}{\sqrt{md}} \tilde{X}_r(X - U(s))^T a_r(s).
\]
Then, one can bound the entire weight matrix as follows:
\[
\frac{d}{ds} W(s) \leq \frac{\|X\|}{\sqrt{md}} \|X - U(s)\| F \|A(s)\| F
\]
\[
\leq \frac{\sqrt{\lambda_n}}{\sqrt{md}} \|X - U(s)\| F \|A(s)\| F
\]
\[
\leq \frac{\sqrt{\lambda_n}(\|A(0)\| + R_a)}{\lambda_n} \exp(-\frac{\lambda_0 s}{2d}) \|X - U(0)\| F.
\]
In the second step, we use the fact $\|CD\|_F \leq \|C\|_F \|D\|_F$ for any matrices $C, D$, and $\|X\|_F = \lambda_n$. The last step follows from $\|A(s)\| \leq \|A(0)\| + R_a$ and Lemma 12. Using the same continuity, we have
\[
\|W(t) - W(0)\|_F \leq \lim_{t' \to t} \int_0^{t'} \frac{d}{ds} W(s) \| F
\]
\[
\leq \lim_{t' \to t} \sqrt{\lambda_n} \|A(0)\| + R_a \|X - U(0)\| F \|A(s)\| F \exp(-\frac{\lambda_0 s}{2d}) \|X - U(0)\| F
\]
\[
\leq 2\sqrt{\lambda_n}(\|A(0)\| + R_a) \|X - U(0)\| F \|A(s)\| F \exp(-\frac{\lambda_0 s}{2d}) \|X - U(0)\| F.
\]
\[
\text{Lemma 14: Fix } t > 0. \text{ Suppose } \lambda_{\min}(K_2(t)) \geq \frac{\lambda_0}{2} \text{ and } \|W(s) - W(0)\|_F \leq R_a \text{ for all } 0 \leq s < t, \text{ then for } r = 1, 2, \ldots, m \text{ we have } \|A(t) - A(0)\|_F \leq R'_a \text{ where}
\]
\[
R'_a \leq 2\sqrt{\lambda_n}(\|W(0)\| + R_a) \|X - U(0)\| F.
\]
Proof: For $s \in [0, t)$, we use the gradient derived in (D.2) and (D.4) to obtain:
\[
\frac{d}{ds} a_r(s) = -\nabla_{a_r} L(W(s), A(s))
\]
\[
= \frac{1}{\sqrt{md}} \sum_{i=1}^{n} \phi(w_i^T x_i)(x_i - u_i(s))
\]
\[
= \frac{1}{\sqrt{md}} (X - U(s)) \phi(X^T w_r(s)).
\]
Then, one can write
\[
\frac{d}{ds} A(s) \| F
\]
\[
= \frac{1}{\sqrt{md}} (X - U(s)) \phi(X^T W(s)) \| F
\]
\[
\leq \frac{\sqrt{\lambda_n}}{\sqrt{md}} \|X - U(s)\| F \|W(s)\| F
\]
\[
\leq \frac{\sqrt{\lambda_n}(\|W(0)\| + R_w)}{\sqrt{md}} \exp(-\frac{\lambda_0 s}{2d}) \|X - U(0)\| F,
\]
where we use $\|X\| \leq \sqrt{n}$, $\|W(s)\| \leq \|W(0)\| + R_w$. The last step follows from Lemma 12. Now, we integrate out $s$
\[
\|A(t) - A(0)\| F \leq \int_0^t \frac{d}{ds} A(s) \| F
\]
\[
\leq \int_0^t \sqrt{\lambda_n}(\|W(0)\| + R_w) \exp(-\frac{\lambda_0 s}{2d}) \|X - U(0)\| F ds
\]
\[
\leq 2\sqrt{\lambda_n}(\|W(0)\| + R_w) \|X - U(0)\| F = R'_a,
\]
which is what we need to prove.

\[
\text{Lemma 15: If } R'_w < R_w \text{ and } R'_a < R_a, \text{ then for all } t \geq 0, \text{ we have}
\]
\[
(i) \lambda_{\min}(K_2(t)) \geq \frac{\lambda_0}{2} \text{ and for all } r \in [m], \|W(t) - W(0)\|_F \leq R'_w, \|A(t) - A(0)\|_F \leq R'_a;
\]
\[
(ii) \text{ If (i) holds, then } \|X - U(t)\|_F \leq \exp(-\frac{\lambda_0}{2d}) \|X - U(0)\|_F.
\]
Proof: Suppose on the contrary that
\[
T = \{ t \geq 0 : \lambda_{\min}(K_2(t)) \leq \frac{\lambda_0}{2} \text{ or } \|W(t) - W(0)\|_F > R'_w \}
\]
\[
\|W(t_0) - W(0)\|_F > R'_w.
\]

This is not an empty set. Therefore, $t_0 \triangleq \inf T$ exists. Using the same continuity argument as in Lemma 7, one can verify that $t_0 > 0$.

First, if $\lambda_{\min}(K_2(t_0)) \leq \frac{\lambda_0}{2}$, then by Lemma 11, $\|W(t) - W(0)\|_F > R'_w$, which is a contradiction because it violates the minimality of $t_0$.

The other two cases are similar, so we will prove one of them. If it holds true that
\[
\|W(t) - W(0)\|_F > R'_w,
\]
then the definitions of $t_0$ and $T$ implies that for any $s \in [0, t_0)$, $\lambda_{\min}(K_2(s)) \geq \frac{\lambda_0}{2}$ and $\|A(s) - A(0)\|_F \leq R'_a$. Then, Lemma 13 leads to:
\[
\|W(t_0) - W(0)\|_F \leq R'_w,
\]
which is a contradiction. Therefore, we have finish the proof.

\[
\text{Proof of Theorem 6: With the results, we can prove the Theorem. From Lemma 15, if } R'_w < R_w \text{ and } R'_a < R_a, \text{ then}
\]
\[
\|X - U(t)\|_F \leq \exp(-\frac{\lambda_0}{d}) \|X - U(0)\|_F^2.
\]

We only need the conditions $R'_w = R'_a \leq R_w = R_a$ to satisfy for this to work. The conditions are
\[
\frac{\lambda_0}{m} (2\|W(0)\| + R_w) R_w \leq \frac{\lambda_0}{4};
\]
and $R_w < R'_w = \frac{2\sqrt{\lambda_n}(\|A(0)\| + R_w) \|X - U(0)\|_F}{\sqrt{md}}$.

Note that $\|X - U(0)\|_F^2 \leq 3n/2d$ with probability at least $1 - \delta$. Also, using a standard bound on sub-Gaussian matrices, we have $\|W(0)\| \leq 2\sqrt{m} + \sqrt{d}$ and $\|A(0)\| \leq 2\sqrt{m} + \sqrt{d}$ with probability at least $1 - 2 \exp(-m)$. Then if we have the order of $m \geq \Omega \left( \frac{\lambda_0^4 \lambda n}{\lambda_0^2 \lambda_n^2} \right)$, Therefore, we finished the proof for the gradient flow theorem.
C. Gradient Descent

As above, we will now appropriately discretize the gradient flow to obtain a convergence result for gradient descent with finite step size for the jointly-trained regime.

**Theorem 9:** Suppose Assumptions 1 and 2 hold. At initialization, suppose the weights are independently drawn from $w_r \sim \mathcal{N}(0, I)$ and $a_r \sim \text{Unif}([-1, 1]^{d})$ for all $r \in [m]$. If $m \geq C\frac{\eta^2 \lambda_0}{\delta}$ for some large enough constant $C$, then with probability at least $1 - \delta$ the gradient descent on $W$ with step size $\eta = \Theta(\frac{\lambda_0}{\sqrt{m}})$,

$$
\|X - U(k)\|_F^2 \leq (1 - \eta \lambda_0/2d)^k \|X - U(0)\|_F^2.
$$

We will prove 7 by induction. The base case when $k = 0$ is trivially true. Assume holds for $k'$, i.e., both hold for $k' = k + 1$. First, we prove that $\|W(k + 1) - W(0)\|_F$ and $\|A(k + 1) - A(0)\|_F$ are small enough, and we then use that to bound $\|X - U(k + 1)\|_F$.

In this section, we define and assume that

$$R_w < \frac{4\sqrt{d}d_{\mathcal{A}_m}(\|A(0)\| + R_a)}{\sqrt{m} \lambda_0} \triangleq R'_w$$

and

$$R_a < \frac{4\sqrt{d}d_{\mathcal{A}_m}(\|W(0)\| + R_w)}{\sqrt{m} \lambda_0} \triangleq R'_a.$$

First, we show the following auxiliary lemma.

**Lemma 16:** If the condition (D.10) holds for $k' = 0, 1, \ldots , k$, then we have

$$\|W(k + 1) - W(0)\|_F \leq R'_w, \text{ and } \|A(k + 1) - A(0)\|_F \leq R'_a$$

with probability at least $1 - \delta$ for any $\delta \in (0, 1)$.

**Proof:** We prove this by induction. Clearly, both hold when $k' = 0$. Assuming that both hold for $k' \leq k$. We will prove both hold for $k' = k + 1$.

We use the expression of the gradients over $w_r$ and $a_r$ in (D.1) and (D.2):

$$\nabla_{w_r} L(W(k), A(k)) = - \sum_{i=1}^{n} \frac{1}{\sqrt{md}} \mathbb{1}_{[w_r(k)^\top x_i \geq 0]} x_i a_r(k)^\top (x_i - u_i(k))$$

and

$$\nabla_{a_r} L(W(k), A(k)) = - \sum_{i=1}^{n} \frac{1}{\sqrt{md}} \phi(w_r(k)^\top x_i)(x_i - u_i(k))$$

$$= - \frac{1}{\sqrt{md}} (X - U(k)) \phi(X^\top w_r(k)).$$

Then, the difference of the weight matrix $W$ is:

$$\|W(k + 1) - W(0)\|_F \leq \eta \|X\| \sum_{k'=0}^{k} \text{vec}(X - U(k')) \|A(0)\| + R_a$$

$$\leq \eta \sqrt{d_{\mathcal{A}_m}(\|A(0)\| + R_a)} \sum_{k'=0}^{k} (1 - \eta \lambda_0/2d)^{k'/2} \|X - U(0)\|_F$$

$$\leq \eta \sqrt{d_{\mathcal{A}_m}(\|A(0)\| + R_a)} \|X - U(0)\|_F \sum_{k'=0}^{\infty} (1 - \eta \lambda_0/2d)^{k'/2}$$

$$= \eta \sqrt{d_{\mathcal{A}_m}(\|A(0)\| + R_a)} \|X - U(0)\|_F \frac{4d}{\sqrt{m} \lambda_0}$$

where the third and fourth step follow from $\|\tilde{X}_r(k')\| \leq \|X\| / \sqrt{\lambda_n}$ and the induction hypothesis $\|A(k')\| \leq \|A(0)\| + R_a$. The last step follows from

$$\sum_{i=0}^{\infty} (1 - \eta \lambda_0/2d)^{i/2} \leq \frac{4d}{\eta \lambda_0}.$$

Similarly, we bound the difference of the weight matrix $A$ between $k + 1$ and 0:

$$\|A(k + 1) - A(0)\|_F$$

$$= \eta \|\sum_{k'=0}^{k} \frac{1}{\sqrt{md}} (X - U(k')) \phi(X^\top W(k'))\|_F$$

$$\leq \eta \|X\| \sum_{k'=0}^{k} \|X - U(k')\|_F \|W(k')\|_F$$

$$\leq \eta \sqrt{d_{\mathcal{A}_m}(\|W(0)\| + R_w)} \|X - U(0)\|_F \sum_{k'=0}^{\infty} (1 - \eta \lambda_0/2d)^{k'/2} \|X - U(0)\|_F (\|W_0\| + R_w)$$

$$= \eta \sqrt{d_{\mathcal{A}_m}(\|W_0\| + R_w)} \|X - U(0)\|_F \frac{4d}{\sqrt{m} \lambda_0}$$

$$= \frac{4\sqrt{d}d_{\mathcal{A}_m}(\|W(0)\| + R_w)}{\sqrt{m} \lambda_0} \|X - U(0)\|_F = R'_a.$$

where the third step and fourth step follow from the facts that $\|\phi(X^\top W(k'))\| \leq \|X\| \|W(k')\|_F$ and $\|X\| = \sqrt{\lambda_n}$, and $\|W(k')\| \leq \|W(0)\| + R_w$.

We have therefore shown that $\|W(k') - w(0)\|_F \leq R'_w$ and $\|A(k') - A(0)\|_F \leq R'_a$ for $k' = k + 1$.

Now, we expand $\|X - U(k + 1)\|_F^2$ in terms of the step $k$.

Recall the update rule:

$$W(k + 1) = W(k) - \eta \nabla W_L(W(k), A(k)), \quad k = 0, 1, \ldots$$

$$A(k + 1) = A(k) - \eta \nabla A_L(W(k), A(k)), \quad k = 0, 1, \ldots$$

where the gradients is given above. Next, we compute the difference of the prediction between two consecutive steps,
a discrete version of $\frac{du_i(t)}{dt}$. For each $i \in [n]$, we have

$$u_i(k+1) - u_i(k) = \frac{1}{\sqrt{md}} \sum_{r=1}^{m} \left( a_r(k+1)\phi(w_r(k+1)^T x_i) - a_r(k)\phi(w_r(k)^T x_i) \right)$$

(D.13)

For a particular $r$, we have $a_r(k+1) = a_r(k) - \eta \nabla_{a_r} L$ and $w_r(k+1) = w_r(k) - \eta \nabla_{w_r} L$, and if the activation pattern does not change, we can write the term inside the sum as:

$$a_r(k+1)\phi(w_r(k+1)^T x_i) - a_r(k)\phi(w_r(k)^T x_i) = -\eta a_r(k)(\nabla_{w_r} L)^T x_i[|w_r(k)^T x_i| \geq 0]$$

$$-\eta(\nabla_{a_r} L)w_r(k)^T x_i[|w_r(k)^T x_i| \geq 0]$$

$$+ \eta^2(\nabla_{a_r} L)(\nabla_{w_r} L)^T x_i[|w_r(k)^T x_i| \geq 0],$$

where the first part corresponds to kernel $G(t)$ and the second part corresponds to the $H(t)$ shown in the gradient flow analysis. With this intuition, we split the right hand side into two parts: $v_{1,i}$ represents the terms that the pattern does not change and $v_{2,i}$ represents the remaining term that pattern may change.

For each $i \in [n]$, we define $S_i = \{ r \in [m] : |w_r(k+1)^T x_i| \geq 0 \} = [1,|w_r(k)^T x_i| \geq 0]$, and $S_i^c = [m] \setminus S_i$. Then, we write $v_{1,i}$ and $v_{2,i}$ as follows:

$$v_{1,i} = \frac{1}{\sqrt{md}} \sum_{r \in S_i} \left( a_r(k+1)\phi(w_r(k+1)^T x_i) - a_r(k)\phi(w_r(k)^T x_i) \right),$$

$$v_{2,i} = \frac{1}{\sqrt{md}} \sum_{r \in S_i^c} \left( a_r(k+1)\phi(w_r(k+1)^T x_i) - a_r(k)\phi(w_r(k)^T x_i) \right).$$

We further write $v_i = (v_{1,1}^T, v_{1,2}^T, \ldots, v_{1,n}^T)^T$ and do the same for $v_2$. Hence, we write

$$\text{vec}(U(k+1) - U(k)) = v_1 + v_2.$$

In order to analyze $v_1 \in \mathbb{R}^n$, we provide definition of $G, G^\perp \in \mathbb{R}^{r \times nd}$ and $H, H^\perp \in \mathbb{R}^{r \times nd}$,

$$G(k)_{i,j} = \frac{1}{m} \sum_{r=1}^{m} x_i^T x_j [w_r(k)^T x_i \geq 0, w_r(k)^T x_j \geq 0] a_r(t)a_r(t)^T,$$

$$G(k)_{i,j}^\perp = \frac{1}{m} \sum_{r=1}^{m} x_i^T x_j [w_r(k)^T x_i \geq 0, w_r(k)^T x_j \geq 0] a_r(t)a_r(t)^T,$$

$$H(k)_{i,j} = \frac{1}{m} \sum_{r=1}^{m} \phi(w_r(t)^T x_i)\phi(w_r(t)^T x_j)I,$$

$$H(k)_{i,j}^\perp = \frac{1}{m} \sum_{r=1}^{m} \phi(w_r(t)^T x_i)\phi(w_r(t)^T x_j)I.$$

Using the fact that $\phi(z) = z \mathbb{I}[z \geq 0]$ and the definition of $S_i$, we expand the forms of the gradients in D.1 and D.2 and get:

$$v_{1,i} = -\frac{1}{\sqrt{md}} \sum_{r \in S_i} \eta a_r(k)(\nabla_{w_r} L)^T x_i[|w_r(k)^T x_i| \geq 0]$$

$$- \frac{1}{\sqrt{md}} \sum_{r \in S_i} \eta(\nabla_{a_r} L)w_r(k)^T x_i[|w_r(k)^T x_i| \geq 0]$$

$$+ \frac{1}{\sqrt{md}} \sum_{r \in S_i} \eta^2(\nabla_{a_r} L)(\nabla_{w_r} L)^T x_i[|w_r(k)^T x_i| \geq 0],$$

$$v_{3,i} = \frac{\eta^2}{\sqrt{md}} \sum_{r \in S_i} (G(k)_{i,j} - G(k)^\perp_{i,j}) + H(k)_{i,j} - H(k)^\perp_{i,j} (x_j - u_j) + v_{3,i},$$

where $v_3$ will be treated as a perturbation:

$$v_{3,i} = \frac{\eta^2}{\sqrt{md}} \sum_{r \in S_i} (\nabla_{w_r} L)(\nabla_{w_r} L)^T x_i[|w_r(k)^T x_i| \geq 0].$$

Then, we can write $v_1$ as:

$$v_1 = \frac{\eta}{d}(K_2(k) - K_2^\perp(k))\text{vec}(X - U(k)) + v_3,$$

in which $K_2(k) = G(k) + H(k)$ — the discrete NTK kernel and $K_2^\perp(k) = H^\perp(k) + G^\perp(k)$. Lastly, we come to the main prediction dynamics in discrete time for vec$(U(k+1)-U(k))$ as:

$$\text{vec}(U(k+1) - U(k)) = \frac{\eta}{d}(K_2(k) - K_2^\perp(k))\text{vec}(X - U(k)) + v_2 + v_3.$$

Using this equation, we can rewrite $\|X - U(k+1)\|^2_F$ in terms of $X - U(k)$ as follows:

$$\|X - U(k+1)\|^2_F = \|\text{vec}(X - U(k+1))\|^2_F$$

$$= \|\text{vec}(X - U(k))\|^2_F - 2\text{vec}(X - U(k))^T\text{vec}(U(k+1) - U(k))$$

$$+ \|U(k+1) - U(k)\|^2_F$$

$$= \|X - U(k)\|^2_F - 2\frac{\eta}{d}\text{vec}(X - U(k))^T K_2(k)\text{vec}(X - U(k))$$

$$- \frac{2\eta}{d}\text{vec}(X - U(k))^T (v_2 + v_3)$$

$$+ \|U(k+1) - U(k)\|^2_F.$$

We define and upper bound each of the following terms

$$C_1 = -\frac{2\eta}{d}\text{vec}(X - U(k))^T K_2(k)\text{vec}(X - U(k)),$$

$$C_2 = \frac{2\eta}{d}\text{vec}(X - U(k))^T K_2^\perp(k)\text{vec}(X - U(k)),$$

$$C_3 = -\frac{2\eta}{d}\text{vec}(X - U(k))^T v_2,$$

$$C_4 = -\frac{2\eta}{d}\text{vec}(X - U(k))^T v_3,$$

$$C_5 = \|U(k+1) - U(k)\|^2_F.$$

Notice that $C_1$ can be upper bounded in terms of $\lambda_{\min}(K_2(k)) \geq \lambda_{\min}(H(k))$, which is ensured as long as
the movement in the weight is sufficiently small (shown in Lemma 16.) $C_2$ can be upper bounded also using the kernel with bound on its spectral norm.

Proof of Theorem 7: We will prove Theorem 7 by induction. The base case when $k = 0$ is trivially true. Assume that the claim holds for $k' = 0, 1, \ldots, k$ and we want to show that (D.10) also holds for $k' = k + 1$. For $k' = k + 1$, we have

$$
\|X - U(k + 1)\|_F^2 = \|X - U(k)\|_F^2 + C_1 + C_2 + C_3 + C_4 + C_5
$$

Now, we invoke the bound for each of these terms from Claims 9, 10, 11, 12 and 13 in Appendix D and Lemma 16. Then, we want to choose $\eta$ and $R_w$ such that

$$
1 - \frac{\eta \lambda_0}{d} + \frac{16 \eta^2}{md} \lambda_0^2 + \frac{8 \eta^2 \sqrt{n} \lambda_0^2}{d} \|X - U(0)\|_F^2 \leq 1 - \frac{\eta \lambda_0}{2d}.
$$

(D.15)

If we set $\eta = \frac{\lambda_0}{8d \sqrt{n}}$ and use $\|X - U(0)\|_F \leq C \sqrt{n}$, we have the two dominating terms are

$$
\frac{8 \eta \lambda_0}{d} \leq \frac{\eta \lambda_0}{8d} \text{, and} \quad \frac{8 \eta^2 \sqrt{n} \lambda_0^2}{d} \|X - U(0)\|_F \leq \eta \lambda_0 \frac{\lambda_0}{8d}.
$$

This implies that

$$
\|X - U(k + 1)\|_F^2 \leq (1 - \frac{\eta \lambda_0}{2d}) \|X - U(k)\|_F^2.
$$

Proof: Since we have proved that $\|W(k) - W(0)\|_F \leq R_w'$, using Lemma 11 with the choice of $R_w < R_w'$, we have

$$
\|H(k) - H(0)\| \leq \frac{\lambda_0}{4}.
$$

Moreover, $G(k)$ is p.s.d, therefore $\lambda_{\text{min}}(K_2(k)) \geq \lambda_{\text{min}}(H(k)) \geq \frac{\lambda_0}{2}$, and as a result,

$$
\text{vec}(X - U(k))^\top K_2(k) \text{vec}(X - U(k)) \geq \frac{\lambda_0}{2} \|X - U(k)\|^2 = \frac{\lambda_0}{2} \|X - U(k)\|_F^2.
$$

and $C_1 \leq - \frac{\eta \lambda_0}{d} \|X - U(k)\|_F^2.

Claim 10: Let $C_2 = \frac{2n}{m} \|\text{vec}(X - U(k))\|^2 K_2(k) \|\text{vec}(X - U(k))\|^2$.

Proof: We need to bound the spectral norm of $K_2(k)$, defined as $K_2(k) = G(k)^1 + H(k)^1$. We will bound their spectral norms. We have

$$
\|G(k)^1\| = \| \frac{1}{m} m \sum_{r=1}^m \text{diag}(I[r \in S^1_r]) \bar{X}_r^\top \tilde{X}_r \otimes a_r(k) a_r(k)^\top \|
\leq \frac{1}{m} \|X\|^2 \| \sum_{r=1}^m a_r(k) a_r(k)^\top I[r \in S^1_r] \|
\leq \frac{\lambda_0}{m} \|A(0)\|^2
\leq \frac{4 \lambda_0}{m} \|A(0)\|^2,
$$

where we use the assumption $R_w \leq \|A(0)\|$. Similarly, using $R_w \leq \|W(0)\|$ we have

$$
\|H(k)^1\|
= \| \frac{1}{m} m \sum_{r=1}^m \text{diag}(I[r \in S^1_r]) \phi(X^\top w_r(k)) \phi(w_r(k)^\top X) \otimes I \|
\leq \frac{1}{m} \|X\|^2 \|W(0)\|^2
\leq \frac{4 \lambda_0}{m} \|W(0)\|^2.
$$

Moreover, using a standard bound on sub-Gaussian matrices, we have $\|W(0)\| \leq 2 \sqrt{m + d}$ and $\|A(0)\| \leq 2 \sqrt{m + d}$ with probability at least $1 - 2 \exp(-m)$. Then,

$$
C_2 = \frac{2n}{m} \|\text{vec}(X - U(k))\|^2 K_2(k) \|\text{vec}(X - U(k))\|^2
\leq \frac{8 \eta \lambda_0}{d} \|X - U(k)\|_F^2
$$

with probability at least $1 - 2 \exp(-m)$.

Proof: We re-state the results in the proof in Lemma 13 and Lemma 14 to bound the remaining terms $C_{3}, C_{4}, C_{5}$:

$$
\|\nabla W L(W(k), A(k))\|_F \leq \frac{\sqrt{\lambda_0}}{\sqrt{md}} \|X - U(k)\|_F \|A(k)\|.
$$

(D.16)

$$
\|\nabla A L(W(k), A(k))\|_F \leq \frac{\sqrt{\lambda_0}}{\sqrt{md}} \|X - U(k)\|_F \|W(k)\|.
$$

(D.17)
Claim 11: Let \( C_3 = -\frac{2\eta}{d} \text{vec}(X - U(k))^\top v_2 \). We have
\[
C_3 \leq \frac{16\eta^2}{d} \sqrt{n\lambda_n} \|X - U(k)\|_F^2.
\]
with probability at least \( 1 - 3 \exp(-m) \).

Proof: We have \( \text{vec}(X - U(k))^\top v_2 \leq \|v_2\|_F \|X - U(k)\|_F \), so we need to bound \( \|v_2\|_F \). Let \( D_i = \text{diag}(I[1 \in S_i^+], \ldots, I[m \in S_i^+]) \), then:
\[
\|v_2\|^2 = \sum_{i=1}^n \|v_{2,i}\|^2
\]
\[
= \frac{1}{md} \sum_{i=1}^n \left\| A(k+1) D_i \phi(W(k+1)^\top x_i) - A(k) D_i \phi(W(k)^\top x_i) \right\|^2
\]
\[
\leq \frac{2\eta^2}{md} \sum_{i=1}^n \left\| (\nabla_{W_{L}})(D_i \phi(W(k+1)^\top x_i)) \right\|^2
\]
\[
+ \left\| A(k) D_i (\nabla_{W_{L}} L)^\top x_i) \right\|^2
\]
\[
\leq \frac{2\eta^2}{md} \left\| W(k+1)^\top x_i \right\|^2 + \frac{2n\lambda_n^2}{m} \|X - U(k)\|_F^2 \|W(k)\|_F^2 + \|A(k)\|_F^2
\]
\[
\leq \frac{64n\lambda_n\eta^2}{md} \|X - U(k)\|_F^2,
\]
with probability at least \( 1 - 3 \exp(-m) \).

Claim 12: Let \( C_4 = -\frac{2\eta}{d} \text{vec}(X - U(k))^\top v_3 \). We have
\[
C_4 \leq \frac{8\eta^2}{d} \sqrt{n\lambda_n^2} \|X - U(k)\|_F \|X - U(0)\|_F
\]
with probability at least \( 1 - 2 \exp(-m) \).

Proof: We have \( \text{vec}(X - U(k))^\top v_3 \leq \|v_3\|_F \|X - U(k)\|_F \). We want to bound \( \|v_3\|_F \). Let \( D_i = \text{diag}(I[w_1(k)^\top x_i \geq 0], \ldots, I[w_m(k)^\top x_i \geq 0]) \)
\[
\|v_3\|^2 = \sum_{i=1}^n \|v_{3,i}\|^2
\]
\[
= \sum_{i=1}^n \left\| \frac{\eta^2}{\sqrt{md}} \sum_{r \in S_i} (\nabla_{a,r} L)(\nabla_{w_r} L)^\top x_i \right\|^2
\]
\[
\leq \frac{\eta^4}{md} \left\| \nabla_A L \right\|^2 \left\| \nabla_{W_{L}} D_i^\top x_i \right\|^2
\]
\[
= \frac{\eta^4}{md} \left\| \nabla_A L \right\|^2 \|\nabla_{W_{L}} D_i^\top x_i \|^2
\]
\[
\leq \frac{\eta^4 n\lambda_n^2}{md^2} \|X - U(k)\|_F^2 \|A(k)\|_F^2 \|W(k)\|^2
\]
\[
\leq \frac{16\eta^2 n\lambda_n^2}{d^2} \|X - U(k)\|_F^2,
\]
with probability at least \( 1 - 2 \exp(-m) \). Therefore,
\[
C_3 \leq \frac{8\eta^2}{d} \sqrt{n\lambda_n^2} \|X - U(k)\|_F^2 \|X - U(0)\|_F
\]
\[
\leq \frac{8\eta^2}{d} \sqrt{n\lambda_n^2} \|X - U(k)\|_F \|X - U(0)\|_F
\]
with probability at least \( 1 - 3 \exp(-m) \). Therefore,
\[
||U(k+1) - U(k)||_F^2
\]
\[
= \frac{1}{m^2} \|A(k+1) \phi(W(k+1)^\top X) - A(k) \phi(W(k)^\top X)\|_F^2
\]
\[
\leq \frac{2\eta^2}{m^2} \left( \|\nabla_A L\|_F \|W(k+1)^\top X\|_F \right)^2
\]
\[
+ \|A(k)\|^2 \|\nabla_{W_{L}} L)^\top X\|_F^2
\]
\[
\leq \frac{2\eta^2}{m^2} \left( \|\nabla_{W_{L}} L)^\top X\|_F \right)^2
\]
\[
\leq \frac{64\eta^2 \lambda_n^2}{md} \|X - U(k)\|_F^2,
\]
with probability at least \( 1 - 3 \exp(-m) \).
\[
(1 - \sigma^2) ||x||^2 + \frac{m - 1}{4m} ||\sigma x||^2 + \frac{1}{m} E_w[\phi(x^T x) ||w||^2].
\] (E.3)

Now, we compute the last term:

\[
E_w[\phi(x^T x) ||w||^2] = \frac{d + 2}{2} ||\sigma x||^2.
\] (E.4)

Due to the normalization \( ||x|| = 1 \), we can also write \( \frac{1}{x^T w} \) such that \( x^T v = 0 \), then \( u = \langle w, x \rangle \sim N(0, 1) \) and \( v \sim N(0, 1 - x^T x) \) are conditionally independent given \( x \). Note that since the conditional distribution of \( u \) is unchanged with respect to \( x \), this implies that \( u \) is independent of \( x \); as a result, \( u \) and \( v \) are (independently) uncorrelated.

Also, denote \( \alpha_q = E_{z \sim N(0,1)}[z^T \mathbb{I}(z \geq 0)] \) for the exact value. Using Stein’s Lemma, we can compute the exact values:

\[
\alpha = E_w[u \mathbb{I}(u \geq 0)] = E_{z \sim N(0,1)}[z \mathbb{I}(z \geq 0)] = \frac{1}{\sqrt{2\pi}}, \quad \beta = E_w[u z \mathbb{I}(u \geq 0)] = \frac{1}{\sqrt{2\pi}}, \quad \gamma = E_w[u^2 z \mathbb{I}(u \geq 0)] = \frac{1}{2},
\]

which are all positive. Write \( \phi(z) = \max(0, z) = \mathbb{I}(z \geq 0), \) and

\[
w \sim N(0,1)[\phi(x^T x) ||w||^2] = E_w[\mathbb{I}(\langle w, x \rangle \geq 0) \langle w, x \rangle^2 ||w||^2] = E_w[\mathbb{I}(u \geq 0) u^2 (u^2 + ||v||^2)] (||x|| = 1)
\]

\[
= E_w[\mathbb{I}(u \geq 0) u^4] + E_w[\mathbb{I}(u \geq 0) u^2] E_w[||v||^2] = \frac{d + 2}{2}.
\] (E.5)

Changing variables by scaling the variance:

\[
w \sim N(0,\sigma^2 1)[\phi(x^T x) ||w||^2] = \sigma^4 E_w \sim N(0,1)[\phi(x^T x) ||w||^2] = (2d + 4)||\sigma x||^2.
\]

Combining with (E.3):

\[
E_w \sim N(0,\sigma^2 1)[||x - u||^2] = \left( \frac{\sigma^2}{2} - 1 \right)^{\frac{1}{2}} ||x||^2 + \frac{(2d + 3)||\sigma^2 x||^2}{4m}.
\] (E.6)

The second result directly follows from the Lemma with the specific values of \( ||x|| \) and \( \sigma \) plugged in. ■

APPENDIX F

INDUCTIVE BIAS OF INFINITELY WIDE TWO-LAYER AUTOENCODERS

We derive the specific form of \( K_0(x, X) \) and \( K_0 \) in Section III-B for the inductive bias of one-sample and multiple-sample training. In Appendix D-B, we obtained \( du/dt \) for the reconstruction \( u \) of the input \( x \) as:

\[
df_t(x) \frac{dt}{dt} = -\frac{1}{\sqrt{md}} \sum_{j=1}^{n} \sum_{r=1}^{m} \mathbb{I}[w_r^T x \geq 0, w_r^T x_j \geq 0] x^T x_j a_r a_r^T (x_j - u_j) - \frac{1}{\sqrt{md}} \sum_{j=1}^{n} \sum_{r=1}^{m} \phi(w_r^T x) \phi(w_r^T x_j) (x_j - u_j).
\]

And, recall that

\[
K_0(x, X) = \frac{\partial f_0}{\partial \theta} \left[ \left( \frac{\partial f_0}{\partial \theta}(x_1) \right)^T, \ldots, \left( \frac{\partial f_0}{\partial \theta}(x_n) \right)^T \right].
\]

Therefore, we have

\[
K_0 = \frac{1}{md} \sum_{r=1}^{m} \tilde{X}_r(0)^T \tilde{X}_r(0) \otimes a_r a_r^T + \frac{1}{m} \sum_{r=1}^{m} \phi(x^T x_r) \phi(w_r^T x) X \otimes I.
\]

In the limit when \( m \to \infty \), we have:

\[
K_0 = \frac{1}{md} \sum_{r=1}^{m} \mathbb{I}(a_r(0) a_r^T + \phi(w_r^T x) x^T) I \to \frac{1}{d} E[w_r^T x \geq 0] a_r(0) a_r^T + (w_r^T x) x^T I = I/d,
\]

since \( w_r(0), a_r(0) \) are independent and \( w_r^T x \sim N(0, 1) \). Therefore, the reconstruction is governed by the similarity between \( x^T \) and \( x \) via the kernel function. Specifically,

\[
\mu_t(x^t) \to d \cdot K_0(x^t, x)(1 - e^{-t/d}) x,
\]

\[
\gamma_t(x^t) \to f_0(x^t) - d \cdot K_0(x^t, x)(1 - e^{-t/d}) f_0(x).
\]

Moreover, as \( m \to \infty \),

\[
K_0(x^t, x) = \frac{\partial f_0}{\partial \theta} \left[ \frac{\partial f_0}{\partial \theta}(x) \right]^T \to \frac{1}{d} E[w_r^T x \geq 0] x^T \phi(w_r^T x) \phi(w_r^T x) I
\]

\[
= (x^T, x) \frac{\pi - \arccos((x^T, x))}{\pi d} \cdot I + \frac{1}{2\sqrt{d}} I - (x^T, x)^T I.
\]

The last equality is calculated using standard probability arguments.

REFERENCES

[1] Y. Li and Y. Liang, “Learning overparameterized neural networks via stochastic gradient descent on structured data,” in Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 8157–8166.

[2] Z. Allen-Zhu, Y. Li, and Z. Song, “A convergence theory for deep learning via over-parameterization,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 242–252.

[3] D. Zhou, Y. Cao, D. Zhou, and Q. Gu, “Stochastic gradient descent optimizes over-parameterized deep ReLU networks,” 2018, arXiv:1811.08888. [Online]. Available: http://arxiv.org/abs/1811.08888

[4] S. S. Du, X. Zhai, B. Poczos, and A. Singh, “Gradient descent provably optimizes overparameterized neural networks,” in Proc. Int. Conf. Learn. Represent., 2019, pp. 1–19.

[5] S. Oymak and M. Soltanolkotabi, “Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks,” IEEE J. Sel. Areas Inf. Theory, vol. 1, no. 1, pp. 84–105, May 2020.

[6] S. Arora, S. Du, W. Hu, Z. Li, and R. Wang, “Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 322–332.

[7] D. Zhou and Q. Gu,”An improved analysis of training over-parameterized deep neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 1–10.

[8] A. Jacot, F. Gabriel, and C. Hongler, “Neural tangent kernel: Convergence and generalization in neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 8571–8580.

[9] B. A. Olshausen and D. J. Field, “Sparse coding with an overcomplete basis set: A strategy employed by v1,” Vis. Res., vol. 37, no. 23, pp. 3311–3325, Dec. 1997.

[10] S. Dasgupta and L. Schulman, “A probabilistic analysis of EM for mixtures of separated, spherical Gaussians,” J. Mach. Learn. Res., vol. 8, pp. 203–226, Feb. 2007.

[11] C. Zhang, S. Bengio, M. Hardt, and Y. Singer, “Identity crisis: Memorization and generalization under extreme overparameterization,” in Proc. Int. Conf. Learn. Represent., 2020, pp. 1–39. [Online]. Available: https://openreview.net/pdf?id=B16y0VFPr
[12] A. Radhakrishnan, M. Belkin, and C. Uhler, “Memorization in overparameterized autoencoders,” Tech. Rep., 2019.

[13] R.-D. Buhai, Y. Halpern, Y. Kim, A. Risteski, and D. Sontag, “Empirical study of the benefits of overparameterization in learning latent variable models,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 1211–1219. [Online]. Available: https://openreview.net/group?id=ICML.cc/2019/Workshop/Deep_Phenomena

[14] A. Rangamani et al., “Sparse coding and autoencoders,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), Jun. 2018, pp. 36–40.

[15] T. V. Nguyen, R. K. W. Wong, and C. Hegde, “On the dynamics of gradient descent for autoencoders,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 2858–2867.

[16] S. Arora, R. Ge, T. Ma, and A. Moitra, “Simple, efficient, and neural algorithms for sparse coding,” in Proc. Conf. Learn. Theory, 2015, pp. 113–149.

[17] T. V. Nguyen, R. K. W. Wong, and C. Hegde, “A provable approach for double-sparse coding,” in Proc. Conf. Amer. Assoc. Artif. Intell., Feb. 2018, pp. 3852–3859.

[18] J. Lee, L. Xiao, S. S. Schoenholz, Y. Bahri, J. Sohl-Dickstein, and J. Pennington, “Wide neural networks of any depth evolve as linear models under gradient descent,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 1–12. [Online]. Available: https://papers.nips.cc/paper/2019/file/0d1a9651497a38d8b1c3871c84528bd4-Paper.pdf

[19] S. Arora, S. S. Du, W. Hu, Z. Li, R. Salakhutdinov, and R. Wang, “On exact computation with an infinitely wide neural net,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 1–10.

[20] B. Xie, Y. Liang, and L. Song, “Diverse neural network learns true target functions,” in Proc. Artif. Intell. Statist., Apr. 2017, pp. 1216–1224.

[21] R. Tsuchida, F. Roosta, and M. Gallagher, “Invariance of weight distributions in rectified MLPs,” in Proc. Int. Conf. Mach. Learn., 2018, pp. 4995–5004.

[22] S. Du, I. Lee, H. Li, L. Wang, and X. Zhai, “Gradient descent finds global minima of deep neural networks,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 1675–1685.

[23] P. Li and N. Phan-Minh, “On random deep weight-tied autoencoders: Exact asymptotic analysis, phase transition, and implications to training,” in Proc. Int. Conf. Learn. Represent., 2019, pp. 1–32.

[24] Z. Song and X. Yang, “Quadratic suffices for over-parametrization via matrix chernoff bound,” 2019, arXiv:1906.03593. [Online]. Available: http://arxiv.org/abs/1906.03593

[25] J. A. Tropp, “User-friendly tail bounds for sums of random matrices,” Found. Comput. Math., vol. 12, no. 4, pp. 389–434, Aug. 2012.

[26] K. Zhong, Z. Song, P. Jain, P. L. Bartlett, and I. S. Dhillon, “Recovery guarantees for one-hidden-layer neural networks,” in Proc. Int. Conf. Mach. Learn., 2017, pp. 4140–4149.

Thanh V. Nguyen received the B.Sc. degree (Hons.) in computer engineering from the Hanoi University of Science and Technology, Vietnam, in 2012, and the Ph.D. degree in electrical engineering from Iowa State University, Ames, IA, USA, in 2020. He is currently an Applied Scientist with Amazon AWS AI. His research interests include unsupervised learning, learning theory, and deep learning for software engineering.

Raymond K. W. Wong received the B.Sc. (Hons.) and M.Phil. degrees in statistics from The Chinese University of Hong Kong, Hong Kong, in 2008 and 2010, respectively, and the Ph.D. degree in statistics from the University of California at Davis in 2014. He is currently an Associate Professor with the Department of Statistics, Texas A&M University. His research interests include nonparametric and semiparametric regression, low-rank modeling, and statistical learning.

Chinmay Hegde (Senior Member, IEEE) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Madras, in 2006, and the M.S. and Ph.D. degrees in electrical and computer engineering from Rice University, in 2010 and 2012, respectively. He is currently an Assistant Professor of Computer Science and Engineering with New York University. His research interests include theoretical tools and applications of machine learning and signal processing.