Understanding the Role of Nonlinearity in Training Dynamics of Contrastive Learning

Yuandong Tian
Meta AI (FAIR)
yuandong@meta.com

Abstract

While the empirical success of self-supervised learning (SSL) heavily relies on the usage of deep nonlinear models, existing theoretical works on SSL understanding still focus on linear ones. In this paper, we study the role of nonlinearity in the training dynamics of contrastive learning (CL) on one and two-layer nonlinear networks with homogeneous activation \( h(x) = h'(x)x \). We have two major theoretical discoveries. First, the presence of nonlinearity can lead to many local optima even in 1-layer setting, each corresponding to certain patterns from the data distribution, while with linear activation, only one major pattern can be learned. This suggests that models with lots of parameters can be regarded as a brute-force way to find these local optima induced by nonlinearity. Second, in the 2-layer case, linear activation is proven not capable of learning specialized weights into diverse patterns, demonstrating the importance of nonlinearity. In addition, for 2-layer setting, we also discover global modulation: those local patterns discriminative from the perspective of global-level patterns are prioritized to learn, further characterizing the learning process. Simulation verifies our theoretical findings.

1 Introduction

Over the last few years, deep models have demonstrated impressive empirical performance in many disciplines, not only in supervised but also in recent self-supervised setting (SSL), in which models are trained with a surrogate loss (e.g., predictive (Devlin et al., 2018; He et al., 2021), contrastive (Chen et al., 2020; Caron et al., 2020; He et al., 2020) or noncontrastive loss (Grill et al., 2020; Chen & He, 2020)) and its learned representation is then used for downstream tasks.

From the theoretical perspective, understanding the roles of nonlinearity in deep neural networks is one critical part of understanding how modern deep models work. Currently, most works focus on linear variants of deep models (Jacot et al., 2018; Arora et al., 2019a, 2019b; Kawaguchi, 2016; Jing et al., 2022; Tian et al., 2021; Wang et al., 2021). When nonlinearity is involved, deep models are often treated as richer families of black-box functions than linear ones (Arora et al., 2019b, HaoChen et al., 2021). The role played by nonlinearity is also studied, mostly on model expressibility (Gühring et al., 2020; Raghu et al., 2017; Lu et al., 2017) in which specific weights are found to fit the complicated structure of the data well, regardless of the training algorithm. However, many questions remain open: if model capacity is the key, why traditional models like k-NN (Fix & Hodges, 1951) or kernel SVM (Cortes & Vapnik, 1995) do not achieve comparable empirical performance, even if theoretically they can also fit any functions (Hammer & Gersmann, 2003; Devroye et al., 1994). Moreover, while traditional ML theory suggests carefully controlling model capacity to avoid overfitting, large neural models often generalize well in practice (Brown et al., 2020; Chowdhery et al., 2022).

In this paper, we study the critical role of nonlinearity in the training dynamics of contrastive learning (CL). Specifically, by extending the recent α-CL framework (Tian, 2022) and linking it to kernels (Paulsen & Raghupathi, 2016), we show that even with 1-layer nonlinear networks, nonlinearity plays a critical role by creating many local optima. As a result, the more nonlinear nodes in 1-layer networks with different initialization, the more local optima are likely to be collected as learned patterns in the trained weights, and the richer the resulting representation becomes. Moreover, popular loss functions like InfoNCE tends to have more local optima than quadratic ones. In contrast, in the linear setting, contrastive learning becomes PCA under certain conditions (Tian, 2022), and
only the most salient pattern (i.e., the maximal eigenvector of the data covariance matrix) is learned while other less salient ones are lost, regardless of the number of hidden nodes.

Based on this finding, we extend our analysis to 2-layer ReLU setting with non-overlapping receptive fields. In this setting, we prove the fundamental limitation of linear networks: the gradients of multiple weights at the same receptive field are always co-linear, preventing diverse pattern learning.

Finally, we also characterize the interaction between layers in 2-layer network: while in each receptive field, many patterns exist, those contributing to global patterns are prioritized to learn by the training dynamics. This global modulation changes the eigenstructure of the low-level covariance matrix so that relevant patterns are learned with higher probability.

In summary, through the lens of training dynamics, we discover unique roles played by nonlinearity which linear activation cannot do: (1) nonlinearity creates many local optima for different patterns of the data, and (2) nonlinearity enables weight specialization to diverse patterns. In addition, we also discover a mechanism for how global pattern prioritizes which local patterns to learn, shedding light on the role played by network depth. Preliminary experiments on simulated data verify our findings.

Related works. Many works analyze network at initialization (Hayou et al., 2019; Roberts et al., 2021) and avoid the complicated training dynamics. Previous works (Wilson et al., 1997; Li & Yuan, 2017; Tian et al., 2019; Tian, 2017; Allen-Zhu & Li, 2020) that analyze training dynamics mostly focus on supervised learning. Different from (Saunshi et al., 2022; Ji et al., 2021) that analyzes feature learning process in linear models of CL, we focus on the critical role played by nonlinearity. Our analysis is also more general than (Li & Yuan, 2017) that focuses on 1-layer ReLU network with symmetric weight structure trained on sparse linear models. Along the line of studying dynamics of contrastive learning, Jing et al. (2022) analyzes dimensional collapsing on 1 and 2 layer linear networks. Tian (2022) proves that such collapsing happens in linear networks of any depth and further analyze ReLU scenarios but with strong assumptions (e.g., one-hot positive input). Our work uses much more relaxed assumptions and performs in-depth analysis for homogeneous activations.

2 Problem Setup

Notation. In this section, we introduce our problem setup of contrastive learning. Let \( x_0 \sim p_D(\cdot) \) be a sample drawn from the dataset, and \( x \sim p_{aug}(\cdot|x_0) \) be a augmentation view of the sample \( x_0 \). Here both \( x_0 \) and \( x \) are random variables. Let \( f = f(x; \theta) \) be the output of a deep neural network that maps input \( x \) into some representation space with parameter \( \theta \) to be optimized. Given a batch of size \( N \), \( x_0[i] \) represent \( i \)-th sample (i.e., instantiation) of corresponding random variables, and \( x[i] \) and \( x[i'] \) are two of its augmented views. Here \( x[i] \) has \( 2N \) samples, \( 1 \leq i \leq N \) and \( N + 1 \leq i' \leq 2N \).

Contrastive learning (CL) aims to learn the parameter \( \theta \) so that the representation \( f \) are distinct from each other: we want to maximize squared distance \( d_{ij}^2 := \| f[i] - f[j] \|^2/2 \) between samples \( i \neq j \) and minimize \( d_i^2 := \| f[i] - f[i'] \|^2/2 \) between two views \( x[i] \) and \( x[i'] \) from the same sample \( x_0[i] \).

Many objectives in contrastive learning have been proposed to combine these two goals into one. For example, InfoNCE (Oord et al., 2018) minimizes the following (here \( \tau \) is the temperature):

\[
\mathcal{L}_{nce} := -\tau \sum_{i=1}^{N} \log \frac{\exp(-d_i^2/\tau)}{\exp(-d_i^2/\tau) + \sum_{j \neq i} \exp(-d_{ij}^2/\tau)}
\]

In this paper, we follow \( \alpha \)-CL (Tian, 2022) that proposes a general CL framework that covers a broad family of existing CL losses. \( \alpha \)-CL maximizes an energy function \( E_\alpha(\theta) \) using gradient ascent:

\[
\theta_{t+1} = \theta_t + \eta \nabla_{\theta} E_{sg(\alpha(\theta))}(\theta),
\]

where \( \eta \) is the learning rate, \( sg(\cdot) \) is the stop gradient operator, the energy function \( E_\alpha(\theta) := \frac{1}{2} \text{tr} \mathbb{C}_\alpha[f, f] \) and \( \mathbb{C}_\alpha[\cdot, \cdot] \) is the contrastive covariance (Tian, 2022; Jing et al., 2023)

\[
\mathbb{C}_\alpha[a, b] := \frac{1}{2N^2} \sum_{i,j=1}^{N} \alpha_{ij} \left[ (a[i] - a[j])(b[i] - b[j])^T \right] \quad (3)
\]
One important quantity is the pairwise importance \(\alpha(\theta) = [\alpha_{ij}(\theta)][N]_{i,j=1}^N\), which are \(N^2\) weights on pairwise pairs of \(N\) samples in a batch. Intuitively, these weights make the training focus more on hard negative pairs, i.e., distinctive sample pairs that are similar in the representation space but are supposed to be separated away. Many existing CL losses (InfoNCE, triplet loss, etc.) are special cases of \(\alpha\)-CL (Tian, 2022) by choosing different \(\alpha(\theta)\), e.g., quadratic loss corresponds to \(\alpha_{ij} = \text{const}\) and InfoNCE (with \(\epsilon = 0\)) corresponds to \(\alpha_{ij} = \exp(-d_{ij}^2/\tau)/\sum_{j \neq i} \exp(-d_{ij}^2/\tau)\).

For brevity \(C_{\alpha}[x] := C_{\alpha}[x, x]\). For the energy function \(E_{\alpha}(\theta) := \text{tr}C_{\alpha}[f(x; \theta)]\), in this work we mainly study its landscape, i.e., existence of local optima, their local properties and overall the objective function, a winner-take-all effect that neglects many patterns in the data.

Now let us first consider 1-layer network with \(k\) hidden nodes: \(f(x; \theta) = h(W x)\), where \(W = [w_1, \ldots, w_K]^\top \in \mathbb{R}^{K \times d}\), \(\theta = \{W\}\) and \(h(x)\) is the activation. The \(k\)-th row of \(W\) is a weight \(w_k\) and its output is \(f_k := h(w_k^\top x)\). In this case, \(\text{tr}C_{\alpha}[f] = \sum_{k=1}^K C_{\alpha}[f_k]\). We consider per-filter normalization \(\|w_k\|_2 = 1\), which can be achieved by imposing BatchNorm (Ioffe & Szegedy, 2015) at each node \(k\) (Tian, 2022). In this case, optimization can be decoupled into each filter \(w_k\):

\[
\max_{\theta} E_{\alpha}(\theta) = \frac{1}{2} \max_{\|w_k\|_2 = 1, 1 \leq k \leq K} \text{tr}C_{\alpha}[f] = \frac{1}{2} \sum_{k=1}^K \max_{\|w_k\|_2 = 1} C_{\alpha}[h(w_k^\top x)]
\]

Now let’s think about, which parameters \(w_k\) maximizes the summation? For the linear case, since \(C_{\alpha}[h(w^\top x)] = C_{\alpha}[w^\top x] = w^\top C_{\alpha}[x]w\), all \(w_k\) converge to the maximal eigenvector of \(C_{\alpha}[x]\) (a constant matrix), regardless of how they are initialized and what the distribution of \(x\) is. Therefore, the linear case will only learn the most salient single pattern due to the (overly-smooth) landscape of the objective function, a winner-take-all effect that neglects many patterns in the data.
In contrast, nonlinearity can change the landscape and create more local optima in $C_{\alpha}[h(w^\top x)]$, each capturing one pattern. In this paper, we consider a general category of nonlinearity activations:

**Assumption 1** (Homogeneity ([Du et al. 2018](#)) / Reversibility ([Tian et al. 2020](#))), The activation satisfies $h(x) = h'(x)x$.

Many activations satisfy this assumption, including linear, ReLU, LeakyReLU and monomial activations like $h(x) = x^p$ (with an additional global constant). In this case we have:

$$h(w^\top x) = w^\top h'(w^\top x)x = w^\top \tilde{x}^w,$$

where $\tilde{x}^w := x \cdot h'(w^\top x)$ is the input data after nonlinear gating. When there is no ambiguity, we just write $\tilde{x}^w$ as $\tilde{x}$ and omit the weight superscript. One property is $C_{\alpha}[h(w^\top x)] = w^\top C_{\alpha}[\tilde{x}^w]w$.

Now let $A(w) := C_{\alpha}[\tilde{x}^w]$. With the constraint $\|w\|_2 = 1$, the learning dynamics is:

**Lemma 3** (Training dynamics of 1-layer network with homogeneous activation in contrastive learning). The gradient dynamics of Eqn. 5 is (note that $\alpha$ is treated as an independent variable):

$$w_k = P_{w_k}^\perp A(w_k)w_k$$

(7)

Here $P_{w_k}^\perp := I - w_kw_k^\top$ projects a vector into the complementary subspace spanned by $w_k$.

See Appendix B.2 for derivations. Now the question is that: what is the critical point of the dynamics and whether they are attractive (i.e., local optima). In linear case, the maximal eigenvector is the only fixed point; in nonlinear case, we are looking for locally maximal eigenvectors, called LME.

**Definition 3** (Locally maximal eigenvector (LME)). $w_s$ is a locally maximal eigenvector of $A(w)$, if $A(w_s)w_s = \lambda_s w_s$, where $\lambda_s = \lambda_{\text{max}}(A(w_s))$ is the distinct maximal eigenvalue of $A(w_s)$.

It is easy to see each LME is a critical point of the dynamics, since $P_{w_s}^\perp A(w_s)w_s = \lambda P_{w_s}^\perp w_s = 0$.

### 3.1 Examples with multiple LMEs in Relu setting

To see why the nonlinear activation leads to many LMEs in Eqn. 7 we first give two examplar generative models of the input $x$ that show Eqn. 7 has multiple critical points, then introduce more general cases. To make the examples simple and clear, we assume the condition of Corollary 1 (no augmentation and large batchsize), and let $\alpha_{ij} \equiv 1$. Notice that $\tilde{x}^w$ is a deterministic function of $x$, therefore $A(w) := C_{\alpha}[\tilde{x}^w] = V[\tilde{x}^w]$. We also use ReLU activation $h(x) = \max(x, 0)$.

Let $U = [u_1, \ldots, u_M]$ be orthonormal bases ($u_m^\top u_m' = I(m = m')$). Here are two examples:

**Latent categorical model.** Suppose $y$ is a categorical random variable taking $M$ possible values, $P[x|y = m] = \delta(x - u_m)$. Then we have (see Appendix B.1 for detailed steps):

$$A(w)|_{w = u_m} := C_{\alpha}[\tilde{x}^w] = V[\tilde{x}^w] = P[y = m] (1 - P[y = m]) u_m u_m^\top$$

(8)

Now it is clear that $w = u_m$ is an LME for any $m$.

**Latent summation model.** Suppose there is a latent variable $y$ so that $x = Uy$, where $y := [y_1, y_2, \ldots, y_M]$. Each $y_m$ is a standardized Bernoulli random variable: $E[y_m] = 0$. 

---

**Figure 1:** Left: Summary of Sec. 3. (a) We analyze the dynamics of one-layer network $h(w^\top x)$ under CL loss (Eqn. 2). (b) With linear activation $h(x) = x$, then there is only one fixed point (PCA direction). (c) Non-linear activation $h(x)$ creates many critical points and a proper choice of pairwise importance $\alpha$ can make them local optima, enabling learning of diverse features. Right: Convergence patterns (iteration $t$ versus iteration discrepancy $\|w(t + 1) - w(t)\|_2$) of Power Iteration (Eqn. PI) in latent summation models, when $\|U^\top U - I\|_2$ is small but non-zero. In this case, Theorem 3 tells there still exist local optima close to each $u_m$. 

---

**Published as a conference paper at ICLR 2023.**
We show that even weight $w$ which has a maximal and distinct eigenvector of $A(w)$ if the following quantity $ho - w$ leads to different LMEs.

In both cases, the presence of ReLU removes the “redundant energy” so that $A(w)$ can focus on specific directions, creating multiple LMEs that correspond to multiple learnable patterns. The two examples can be computed analytically due to our specific choices on nonlinearity $h$ and $\alpha$.

### 3.2 RELATE LMEs TO LOCAL OPTIMA

Once LMEs are identified, the next step is to check whether they are attractive, or stable critical points, or local optima. That is, whether the weights converge into them and stay there during training. For this, some notations are introduced below.

**Notations.** Let $\lambda_i(w)$ be the $i$-th largest eigenvalue of $A(w)$, and $\phi_i(w)$ the corresponding unit eigenvector. $\lambda_{\text{gap}}(w) := \lambda_1(w) - \lambda_2(w)$ the eigenvalue gap. Let $\rho(w)$ be the local roughness measure: $\rho(w)$ is the smallest scalar to satisfy $\|A(v) - A(w)\|_2 \leq \rho(w)\|v - w\|_2 + O(\|v - w\|_2^2)$ in a local neighborhood of $w$. The following theorem gives a sufficient condition for stability of $w$:

**Theorem 1** (Stability of $w^*$). If $w_*$ is a LME of $A(w_*)$ and $\lambda_{\text{gap}}(w_*) > \rho(w_*)$, then $w_*$ is stable.

This shows that lowering roughness measure $\rho(w_*)$ at critical point $w_*$ could lead to more local optima and more patterns to be learned. To characterize such a behavior, we bound $\rho(w_*):$

**Theorem 2** (Bound of local roughness $\rho(w)$ in ReLU setting). If input $\|x\|_2 \leq C_0$ is bounded, $\alpha$ has kernel structure (Def. 2) and batchsize $N \to +\infty$, then $\rho(w_*) \leq \frac{C_0^2 \|\alpha\|_2}{\pi} r(w_*, \alpha)$, where $r(w, \alpha) := \sum_{\tau=0}^{\infty} \sigma^2(\alpha) \max_{x \in \mathcal{B}} p_\tau(x; \alpha)$.

From Thm., the bound critically depends on $r(\alpha)$ that contains the adjusted density $p_\tau(x; \alpha)$ (Def. 2) at the plane $w_\tau^\top x = 0$. This is because a local perturbation of $w_*$ leads to data inclusion/exclusion close to the plane, and thus changes $\rho(w_*)$. Different $\alpha$ leads to different $p_\tau(x; \alpha)$, and thus different upper bound of $\rho(w_*)$, creating fewer or more local optima (i.e., patterns) to learn. Here is an example that shows Gaussian $\alpha$ (see Lemma [1]), whose normalized version is used in InfoNCE, can lead to more local optima than uniform $\alpha$, by lowering roughness bound characterized by $r(w_*, \alpha)$:

**Corollary 2** (Effect of different $\alpha$). For uniform $\alpha_u$ ($\alpha_{ij} := 1$) and 1-D Gaussian $\alpha_g$ ($\alpha_{ij} := \exp(-\|h(w) x_0[i] - h(w^* x_0[j])\|_2^2/2r)$), we have $r(w_*, \alpha_u) = z_0(\alpha_u) r(w_*, \alpha_u)$ with $z_0(\alpha_u) := \int \exp(-\alpha h^2(w_\tau^\top x)/2r) d\tau(x) \leq 1$. As a result, $z_0(\alpha_g) \ll 1$ leads to $r(w_*, \alpha_g) \ll r(w_*, \alpha_u)$. In practice, $z_0(\alpha_g)$ can be exponentially small (e.g., when most data appear on the positive side of the weight $w_*$) and the roughness with Gaussian $\alpha$ can be much smaller than that of uniform $\alpha$, which is presumably the reason why InfoNCE outperforms quadratic CL loss (Tian, 2022).

### 3.3 FINDING CRITICAL POINTS WITH INITIAL GUESS

In the following, we focus on how can we find an LME, when $A(w)$ does not have analytic form. We show that if there is an “approximate eigenvector” of $A(w) := C_\alpha [\tilde{x} w]$, then a real one is nearby.

Let $L$ be the Lipschitz constant of $A(w)$: $\|A(w) - A(w')\|_2 \leq L \|w - w'\|_2$ for any $w, w'$ on the unit sphere $\|w\|_2 = 1$, and the correlation function $c(w) := w^\top \phi_1(w)$ be the inner product between $w$ and the maximal eigenvector of $A(w)$. We can construct a fixed point using Power Iteration (PI) (Golub & Van Loan, 2013), starting from initial value $w = w(0)$:

$$w(t + 1) \leftarrow A(w(t))w(t)/\|A(w(t))w(t)\|_2 \quad \text{(PI)}$$

We show that even $A(w)$ varies over $\|w\|_2 = 1$, the iteration can still converge to a fixed point $w_*$, if the following quantity $\omega(w)$, called irregularity, is small enough.
Then Power Iteration (Eqn. PI) converges to a critical point at each RF (Eqn. 7), even if objective involving \( c \) is only approximately orthogonal (i.e., \( \| U^\top U - I \| \) is not zero but small), and/or the conditions of Corollary hold roughly, then Theorem tells that multiple local optima close to \( u_m \) still exist for each \( m \) (Fig. [I]). We leave it for future work to further relax the condition.

Possible relation to empirical observations. Since there exist many local optima in the dynamics (Eqn. 7), even if objective involving \( u_k \) are identical (Eqn. 5), each \( u_k \) may still converge to different local optima due to initialization. We suspect that this can be a tentative explanation why larger model performs better: more local optima are collected and some can be useful. Other empirical observations like lottery ticket hypothesis (LTH) [Frankle & Carbin, 2019; Morcos et al., 2019; Tian et al., 2019; Yu et al., 2020], recently also verified in CL [Chen et al., 2021], may also be explained similarly. In LTH, first a large network is trained and pruned to be a small subnetwork \( S \), then retraining \( S \) using its original initialization yields comparable or even better performance, while retraining \( S \) with a different initialization performs much worse. For LTH, our explanation is that \( S \) contains weights that are initialized *luckily*, i.e., close to useful local optima and converge to them during training. We leave a thorough empirical study to justify this line of thought for future work.

Given this intuition, it is tempting to study the distribution of local optima of Eqn. 7, their attractive basin \( \text{Basin}(u_\ast) := \{ w : w(0) = w, \lim_{t \to +\infty} w(t) = u_\ast \} \) for each local optimum \( u_\ast \), and the probability of random initialized weights fall into them. Interestingly, *data augmentation* may play an important role, by removing unnecessary local optima with symmetry (see Appendix B.3), focusing the learning on important patterns. Theorem also gives hints. A formal study is left for future work.

## 4 Two-layer setting

Now we understand how 1-layer nonlinearity learns in contrastive learning setting. In practice, many patterns exist and most of them may not be relevant for the downstream tasks. A natural question arises: how does the network prioritizes which patterns to learn? To answer this question, we analyze the behavior of 2-layer nonlinear networks with non-overlapping receptive fields (Fig. 2.a).

**Setting and Notations.** In the lower layer, there are \( K \) disjoint receptive fields (abbreviated as RF) \( \{ R_k \} \), each has input \( x_k \) and \( M \) weight \( w_{km} \in \mathbb{R}^d \) where \( m = 1 \ldots M \). The output of the bottom-layer is denoted as \( f_k, f_k^{km} \) for its \( km \)-th component, and \( f_k[i] \) for \( i \)-th sample. The top layer has weight \( V \in \mathbb{R}^{d_{out} \times KM} \). Define \( S := V^\top V \). As the \((km, k')\) entry of the matrix \( S \), \( S_{km, k'} := \langle S \rangle_{km, k'} = w_{km}^\top v_{k'} \).

At each RF \( R_k \), define \( \hat{x}_{km} \), as an brief notation of gated input \( \hat{x}_{km}^k := x_k \cdot h'(w_{km}^\top x_k) \). Define \( \hat{x}_k := \{ \hat{x}_{k1}, \hat{x}_{k2}, \ldots, \hat{x}_{KM} \} \in \mathbb{R}^{Md} \) as the concatenation of \( \hat{x}_{km} \) and finally \( \hat{x} := \{ \hat{x}_1, \ldots, \hat{x}_K \} \in \mathbb{R}^{KMd} \) is the concatenation of all \( \hat{x}_k \). Similarly, let \( w_k := \{ w_{km} \}_{m=1}^M \in \mathbb{R}^{Md} \) be a concatenation of all \( w_{km} \) in the same RF \( R_k \), and \( \bar{w} := [w_{km}]_{k=1}^K \in \mathbb{R}^{KMd} \) be a column concatenation of all \( w_k \).

Finally, \( P_w^\perp := \text{diag}_{km}[P_w^\perp_{km}] \) is a block-diagonal matrix putting all projections together.
With all the assumptions, we can compute the term $A_k(w_k) := C_\alpha[\tilde{x}_k] = V[\tilde{x}_k]$. Our Assumption 3 is weaker than orthogonal mixture condition in Tian (2022) that is used to analyze CL, which requires the instance of input $x_k[i]$ to have only one positive component.
4.1 Why nonlinearity is critical: Linear activation fails

Since in each RF $R_k$, there are $M$ filters $\{w_{km}\}$, it would be ideal to have one filter to capture one distinct pattern in the covariance matrix $A_k$. However, with linear activation, $\tilde{x}_k = x_k$ and as a result, learning of diverse features never happens, no matter how large $M$ is (proof in Appendix C.4): 

**Theorem 4** (Gradient Colinearity in linear networks). With linear activation, $W$ follows the dynamics:

$$\dot{w}_{km} = s_{km} b_k(W, V)$$

(14)

where $b_k(W, V) := \mathbb{C}_\alpha \left[ x_k, \sum_{k',m} s_{k'm} w_{k'm}^T x_{k'} \right]$ is a linear function w.r.t. $W$. As a result, (1) $\dot{w}_{km}$ are co-linear over $m$, and (2) if $s_{km} \neq 0$, from any critical point with distinct $\{w_{km}\}$, there exists a path of critical points to identical weights ($\dot{w}_{km} = \dot{w}_k$).

This brings about the weakness of linear activation. First, the gradient of $w_{km}$ within one RF $R_k$ during CL training all points towards the same direction $b_k$; Second, even if the critical points $w_{km}$ have any diversity within RF $R_k$, there exist a path for them to converge to identical weights. Therefore, diverse features, even they reside in the data, cannot be learned by the linear models.

4.2 The effect of global modulation in the special case of $C = 2$ and $M = 1$

When $z$ is binary ($C = 2$) with a single weight per RF ($M = 1$), $w_k$’s dynamics has close form. Let $w_k$ represent $w_{k1}$, the only weight at each $R_k$, $\Delta_k := \mathbb{E}[x_k | z = 1] - \mathbb{E}[x_k | z = 0]$. We have:

**Theorem 5** (Dynamics of $w_k$ under conditional independence). When $C = 2$ and $M = 1$, the dynamics of $w_k$ is given by ($s_k^2$ and $\delta_k \geq 0$ are scalars defined in the proof):

$$\dot{w}_k = P_{w_k} \left( s_k^2 A_k(w_k) + \delta_k \Delta_k \Delta_k^T \right) w_k$$

(15)

See proof in Appendix C.5. There are several interesting observations. First, the dynamics are decoupled (i.e., $\dot{w}_k = A_k(W) w_k$) and other $w_{k'}$ with $k' \neq k$ only affects the dynamics of $w_k$ through the matrix $A_k(W)$. Second, while $A_k(w_k)$ contains multiple patterns (i.e., local optima) in $R_k$, the additional term $\Delta_k \Delta_k^T$ as the global modulation from the top level, encourages the model to learn the pattern like $\Delta_k$ which is a discriminative feature that separates the event of $z = 0$ and $z = 1$. Quantitatively:

**Theorem 6** (Global modulation of attractive basin). If the structural assumption holds: $A_k(w_k) = \sum g(u_i^k w_k) u_i u_i^T$ with $g(\cdot) > 0$ a linear increasing function and $\{u_i\}$ orthonormal bases, then for $A_k + c u_i u_i^T$, its attractive basin of $w_k = u_i$ is larger than $A_k$’s for $c > 0$.

Therefore, if $\Delta_k$ is a LME of $A_k$ and $w_k$ is randomly initialized, Thm 5 tells that $P_{w_k \rightarrow \Delta_k}$ is higher than the probability that $w_k$ goes to other patterns of $A_k$, i.e., the global variable $z$ modulates the training of the lower layer. This is similar to “Backward feature correction” (Allen-Zhu & Li 2020) and “top-down modulation” (Tian et al. 2019) in supervised learning, here we show it in CL.

We also analyze how BatchNorm helps alleviates diverse variances among RFs (see Appendix D).
Figure 4: Overall matching score $\bar{\chi}_+^\phi$ (Eqn. [16]) with InfoNCE (top row) and quadratic loss (bottom row). When $P = 1$, linear model works well regardless of the degree of over-parameterization $\beta$, while ReLU requires large over-parameterization to perform well. When each $R_k$ has multiple patterns ($P > 1$) related to generators, ReLU models can capture diverse patterns better than linear ones in the over-parameterization region $\beta > 1$. We found similar trend for other homogeneous activations such as LeakyReLU (with negative slope 0.05) and quadratic. In contrast, linear models are much less affected by over-parameterization. While the trends are similar, quadratic loss is not as effective as InfoNCE in feature learning. Each setting is repeated 3 times and mean/std are reported. See Appendix (Fig. 9 and Fig. 10) for $\bar{\chi}_-^{\phi}$.

Figure 5: Visualization of learned weights with $P = 3$ (3 local patterns related to generators at each RF) and $\beta = 5$ (5x over-parameterization). Each of the $K = 10$ subfigures corresponds to a RF ($R_0, R_k$). In each subfigure, the left panel is the learned weight by ReLU, while the right panel is from linear activations. 15 rows corresponds to $M = \beta P = 15$ weights and each weight is $d = 8$ dimensional. With ReLU activation, learned weights clearly capture the 3 candidate tokens within $R_k^0$ at each RF $R_k$, while linear activation cannot.
5 EXPERIMENTS

Setup. To verify our finding, we perform contrastive learning with a 2-layer network on a synthetic dataset containing token sequences, generated as follows. From a pool of \( G = 40 \) generators, we pick a generator of length \( K \) in the form of \(* * C + B + A + D*\) (here \( K = 10 \)) and generate \( \text{EFCDBAACDB} \) by sampling from \( d = 20 \) tokens for each wildcard \(*\). The final input \( x \) is then constructed by replacing each token \( a \) with the pre-defined embedding \( u_a \in \mathbb{R}^d \). \{ \{ u_a \} \} forms an orthonormal basis (see Fig. 3). The data augmentation is achieved by generating another sequence from the same generator.

While there exists \( d = 20 \) tokens, in each RF \( R_k \) we pick a subset \( R^g_k \) of \( P < d \) tokens as the candidates used in the generator, to demonstrate the effect of global modulation. Before training, each generator is created by first randomly picking 5 receptive fields, then picking one of the \( P \) tokens from \( R^g_k \) at each RF \( R_k \) and filling the remaining RFs with wildcard \(*\). Therefore, if a token appears at \( R_k \) but \( a \notin R^g_k \), then \( a \) must be instantiated from the wildcard. Any \( a \notin R^g_k \) is noise and should not be learned in the weights of \( R_k \) since it is not part of any global pattern from the generator.

We train a 2-layer network on this dataset. The 2-layer network has \( K = 10 \) disjoint RFs, within each RF, there are \( M = \beta P \) filters. Here \( \beta \geq 1 \) is a hyper-parameter that controls the degree of over-parameterization. The network is trained with InfoNCE loss and SGD with learning rate \( 2 \times 10^{-3} \), momentum 0.9, and weight decay \( 5 \times 10^{-5} \) for 5000 minibatches and batchsize 128. Code is in PyTorch runnable on a single modern GPU.

Evaluation metric. We check whether the weights corresponding to each token is learned in the lower layer. At each RF \( R_k \), we know \( R^g_k \), the subsets of tokens it contains, as well as their embeddings \( \{ u_a \} \) corresponding to \( a \in R^g_k \) due to the random process, and verify whether these embeddings are learned after the model is trained. Specifically, for each token \( a \in R^g_k \), we look for its best match on the learned filter \( \{ w_{km} \} \), as formulated by the following per-RF score \( \chi_+(R_k) \) and overall matching score \( \bar{\chi}_+ \) for \( a \notin R^g_k \):

\[
\chi_+(R_k) = \frac{1}{P} \sum_{a \in R^g_k} \max_m \frac{w_{km}^T u_a}{\|w_{km}\| \|u_a\|}, \quad \bar{\chi}_+ = \frac{1}{K} \sum_k \chi_+(R_k)
\]

5.1 RESULTS

Linear vs ReLU activation and the effect of over-parameterization (Sec. 4.1). From Fig. 4 we can clearly see that ReLU (and other homogeneous) activations achieve better reconstruction of the input patterns, when each RF contains many patterns (\( P > 1 \)) and specialization of filters in each RF is needed. On the other hand, when \( P = 1 \), linear activation works better. ReLU activation clearly benefits from over-parameterization (\( \beta > 1 \)): the larger \( \beta \) is, the better \( \bar{\chi}_+ \) becomes. In contrast, for linear activation, over-parameterization does not quite affect the performance, which is consistent with our theoretical analysis.

Quadratic versus InfoNCE. Fig. 4 shows that quadratic CL loss underperforms InfoNCE, while the trend of linear/ReLU and over-parameterization remains similar. According to Corollary 1 non-uniform \( \alpha \) (e.g., Gaussian \( \alpha \), Lemma 1) creates more and deeper local optima that better accommodate local patterns, yielding better performance. This provides a novel landscape point of view on why non-uniform \( \alpha \) is better, expanding the intuition that it focuses more on important sample pairs.

Global modulation (Sec. 4.2). As shown in Fig. 5 the learned weights indeed focus on the token subset \( R^g_k \) that receives top-down support from the generators and no noise token is learned. We also verify that quantitatively by computing \( \bar{\chi}_- \) over multiple runs, provided in Appendix (Fig. 9-10).

REFERENCES

Zeyuan Allen-Zhu and Yuanzhi Li. Backward feature correction: How deep learning performs deep learning. arXiv preprint arXiv:2001.04413, 2020.

Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu. A convergence analysis of gradient descent for deep linear neural networks. In International Conference on Learning Representations, 2019a. URL https://openreview.net/forum?id=SkMqg3C5K7.
Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*, 2019b.

Mary L Boas and Philip Peters. Mathematical methods in the physical sciences, 1984.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

James R Bunch, Christopher P Nielsen, and Danny C Sorensen. Rank-one modification of the symmetric eigenproblem. *Numerische Mathematik*, 31(1):31–48, 1978.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Michael Carbin, and Zhangyang Wang. The lottery tickets hypothesis for supervised and self-supervised pre-training in computer vision models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16306–16316, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2020.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kenshi Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Leverkühn, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Pippin, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayanan Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *CoRR*, abs/2204.02311, 2022. doi: 10.48550/arXiv.2204.02311. URL [https://doi.org/10.48550/arXiv.2204.02311](https://doi.org/10.48550/arXiv.2204.02311).

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Luc Devroye, Laszlo Gyori, Adam Krzyzak, and Gabor Lugosi. On the strong universal consistency of nearest neighbor regression function estimates. *The Annals of Statistics*, 22(3):1371–1385, 1994.

Simon S Du, Wei Hu, and Jason D Lee. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. *arXiv preprint arXiv:1806.00900*, 2018.

Francisco M Fernández. *Introduction to perturbation theory in quantum mechanics*. CRC press, 2000.

Evelyn Fix and Joseph Lawson Hodges. Nonparametric discrimination: consistency properties. *Randolph Field, Texas, Project*, pp. 21–49, 1951.
Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019. URL [https://openreview.net/forum?id=rJl-b3RcF7](https://openreview.net/forum?id=rJl-b3RcF7).

Benyamin Ghojogh, Ali Ghodsi, Fakhri Karray, and Mark Crowley. Reproducing kernel hilbert space, mercer’s theorem, eigenfunctions, nyström method, and use of kernels in machine learning: Tutorial and survey. *arXiv preprint arXiv:2106.08443*, 2021.

Mike B Giles. Collected matrix derivative results for forward and reverse mode algorithmic differentiation. In *Advances in Automatic Differentiation*, pp. 35–44. Springer, 2008.

Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013.

Alain Goriely and Craig Hyde. Finite-time blow-up in dynamical systems. *Physics Letters A*, 250 (4-6):311–318, 1998.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *NeurIPS*, 2020.

Ingo Gühring, Mones Raslan, and Gitta Kutyniok. Expressivity of deep neural networks. *arXiv preprint arXiv:2007.04759*, 2020.

Barbara Hammer and Kai Gersmann. A note on the universal approximation capability of support vector machines. *neural processing letters*, 17(1):43–53, 2003.

Jeff Z HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *NeurIPS*, 2021.

Soufiane Hayou, Arnaud Doucet, and Judith Rousseau. On the impact of the activation function on deep neural networks training. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2672–2680. PMLR, 09–15 Jun 2019. URL [https://proceedings.mlr.press/v97/hayou19a.html](https://proceedings.mlr.press/v97/hayou19a.html).

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*, 2021.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pp. 448–456. PMLR, 2015.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

Wenlong Ji, Zhun Deng, Ryumei Nakada, James Zou, and Linjun Zhang. The power of contrast for feature learning: A theoretical analysis. *arXiv preprint arXiv:2110.02473*, 2021.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. *ICLR*, 2022.

Kenji Kawaguchi. Deep learning without poor local minima. *NeurIPS*, 2016.

Yuanzhi Li and Yang Yuan. Convergence analysis of two-layer neural networks with relu activation. *Advances in neural information processing systems*, 30, 2017.

Zhou Lu, Hongming Pu, Feicheng Wang, Zhigiang Hu, and Liwei Wang. The expressive power of neural networks: A view from the width. *Advances in neural information processing systems*, 30, 2017.
Ari Morcos, Haonan Yu, Michela Paganini, and Yuandong Tian. One ticket to win them all: generalizing lottery ticket initializations across datasets and optimizers. *Advances in neural information processing systems*, 32, 2019.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Vern I Paulsen and Mrinal Raghupathi. *An introduction to the theory of reproducing kernel Hilbert spaces*, volume 152. Cambridge university press, 2016.

Maithra Raghu, Ben Poole, Jon Kleinberg, Surya Ganguli, and Jascha Sohl-Dickstein. On the expressive power of deep neural networks. In *international conference on machine learning*, pp. 2847–2854. PMLR, 2017.

Daniel A. Roberts, Sho Yaida, and Boris Hanin. The principles of deep learning theory. *CoRR*, abs/2106.10165, 2021. URL https://arxiv.org/abs/2106.10165

Nikunj Saunshi, Jordan Ash, Saurbhi Goel, Dipendra Misra, Cyril Zhang, Sanjeev Arora, Sham Kakade, and Akshay Krishnamurthy. Understanding contrastive learning requires incorporating inductive biases. *arXiv preprint arXiv:2202.14037*, 2022.

John Michael Tutill Thompson, H Bruce Stewart, and Rick Turner. Nonlinear dynamics and chaos. *Computers in Physics*, 4(5):562–563, 1990.

Yuandong Tian. An analytical formula of population gradient for two-layered relu network and its applications in convergence and critical point analysis. In *International Conference on Machine Learning*, pp. 3404–3413. PMLR, 2017.

Yuandong Tian. Understanding deep contrastive learning via coordinate-wise optimization. *NeurIPS*, 2022.

Yuandong Tian, Tina Jiang, Qucheng Gong, and Ari Morcos. Luck matters: Understanding training dynamics of deep relu networks. *arXiv preprint arXiv:1905.13405*, 2019.

Yuandong Tian, Lantao Yu, Xinlei Chen, and Surya Ganguli. Understanding self-supervised learning with dual deep networks. *arXiv preprint arXiv:2010.00578*, 2020.

Yuandong Tian, Xinlei Chen, and Surya Ganguli. Understanding self-supervised learning dynamics without contrastive pairs. In *International Conference on Machine Learning*, pp. 10268–10278. PMLR, 2021.

Xiang Wang, Xinlei Chen, Simon S Du, and Yuandong Tian. Towards demystifying representation learning with non-contrastive self-supervision. *arXiv preprint arXiv:2110.04947*, 2021.

Charles L Wilson, James L Blue, and Omid M Omidvar. Training dynamics and neural network performance. *Neural Networks*, 10(5):907–923, 1997.

Haonan Yu, Sergey Edunov, Yuandong Tian, and Ari S. Morcos. Playing the lottery with rewards and multiple languages: lottery tickets in rl and nlp. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1xnXRVFwH
A \ PROOFS

A.1 \ PROBLEM \ SETUP \ (SEC.2)

**Lemma 1** (Gaussian $\alpha$). For any function $g(\cdot)$ that is bounded below, if we use $\alpha_{ij} := \exp(-\|g(x_0[i]) - g(x_0[j])\|^2/2\tau)$ as the pairwise importance, then it has kernel structure (Def. 1).

**Proof.** Since $g(\cdot)$ is bounded below, there exists a vector $v$ so that each component of $g(x) - v$ is always nonnegative for any $x$. Let $y[i] := g(x_0[i]) - v \in \mathbb{R}^d$, then $y[i] \geq 0$ and we have:

$$
\alpha_{ij} = \exp\left(-\frac{\|y[i] - y[j]\|^2}{2\tau}\right) = \exp\left(-\frac{\|y[i]\|^2}{2\tau}\right) \exp\left(-\frac{\|y[j]\|^2}{2\tau}\right) \exp\left(-\frac{y[i]y[j]}{2\tau}\right)
$$

And using Taylor expansion, we have

$$
\exp\left(\frac{y^\top y}{\tau}\right) = 1 + \frac{y^\top y}{\tau} + \frac{1}{2} \left(\frac{y^\top y}{\tau}\right)^2 + \ldots + \frac{1}{k!} \left(\frac{y^\top y}{\tau}\right)^k + \ldots
$$

Let

$$
\tilde{\phi}(y) := \left[\begin{array}{c}
\frac{1}{\sqrt{\tau}} \text{AllChoose}(\tau^{-1/2} y, 2) \\
\ldots \\
\frac{1}{\sqrt{\tau}} \text{AllChoose}(\tau^{-1/2} y, k)
\end{array}\right] \geq 0
$$

be an infinite dimensional vector, where AllChoose($y, k$) is a $d^k$-dimensional column vector that enumerates all possible $d^k$ products $y_iy_j \cdots y_k$, where $1 \leq i_k \leq d$ and $y_i$ is the $i$-th component of $y$. Then it is clear that $\exp(y^\top y)/\tau) = \tilde{\phi}(y^\top y)$ and thus

$$
\alpha_{ij} = \phi^\top(x_0[i])\phi(x_0[j]) = \sum_{l=0}^{+\infty} \phi_l(x_0[i])\phi_l(x_0[j])
$$

which satisfies Def. 1. Here

$$
\phi(x) := \exp\left(-\frac{\|\phi\|^2}{2\tau}\right) \tilde{\phi}(y) = \exp\left(-\frac{\|g(x) - v\|^2}{2\tau}\right) \tilde{\phi}(g(x) - v)
$$

is the infinite dimensional feature mapping for input $x$, and $\phi_l(x)$ is its $l$-th component. \hfill \square

**Lemma 2** (Relationship between Contrastive Covariance and Variance in large batch size). If $\alpha$ satisfies Def. 2 then for any function $g(\cdot)$, $C_\alpha[g(x)]$ is asymptotically PSD when $N \rightarrow +\infty$:

$$
C_\alpha[g(x)] \rightarrow \sum_l \frac{\tau^2 E_{x \sim \mathcal{P}_\alpha}(\cdot|x_0)}{\mathcal{C}_\alpha[g(x)]} [E_{x \sim \mathcal{P}_\alpha}(\cdot|x_0)]
$$

**Proof.** First let

$$
C_{\alpha, \text{intra}}[a, b] := \frac{1}{2N^2} \sum_{i=1}^{N} \sum_{j \neq i} \alpha_{ij} (a[i] - a[j])(b[i] - b[j])^T
$$

$$
C_{\alpha, \text{inter}}[a, b] := \frac{1}{2N} \sum_{i=1}^{N} \left( \frac{1}{N} \sum_{j \neq i} \alpha_{ij} \right) (a[i] - a[i'])(b[i] - b[i'])^T
$$

and $C_{\alpha, \text{total}} := C_{\alpha, \text{inter}} + C_{\alpha, \text{intra}}$. Then we have

$$
C_\alpha[g] = C_{\alpha, \text{intra}}[g] - C_{\alpha, \text{inter}}[g].
$$
With the condition, for the first term \( C^\text{inter}_\alpha [g] \), we have
\[
C^\text{inter}_\alpha [g] = \frac{1}{2N^2} \sum_{ij} K(x_0[i], x_0[j])(g(x[i]) - g(x[j]))(g(x[i]) - g(x[j]))^T
\]  
(26)

When \( N \to +\infty \), we have:
\[
C^\text{inter}_\alpha [g] \to \frac{1}{2} \int K(x_0, y_0)(g(x) - g(y))(g(x) - g(y))^\top \mathbb{P}(x, x_0)\mathbb{P}(y, y_0)dx dy dx_0 dy_0
\]
We integrate over \( x_0 \) and \( y_0 \) first:
\[
\phi_i(x_0) \phi_i(y_0) \mathbb{P}(x_0)\mathbb{P}(y_0) dx_0 dy_0
\]
(27)
\[
\phi_i(x_0) \phi_i(y_0) \mathbb{P}(x_0)\mathbb{P}(y_0) dx_0 dy_0
\]
(29)
We now compute the four terms separately. With the condition that \( K(x_0, y_0) = \sum_i \phi_i(x_0) \phi_i(y_0) \), and the definition of adjusted probability \( \hat{p}_i(x) := \frac{1}{z_i} \phi_i(x) \mathbb{P}(x) \) where \( z_i := \int \phi_i(x)\mathbb{P}(x)dx \), for the first term, we have:
\[
\int \phi_i(x_0) \phi_i(y_0) \mathbb{E}[x_0, gg^\top] \mathbb{P}(x_0)\mathbb{P}(y_0) dx_0 dy_0
\]
(30)
\[
\sum_j z_j^2 \mathbb{E}[x_0, gg^\top] \mathbb{P}(x_0)\mathbb{P}(y_0) dx_0 dy_0
\]
(32)
On the other hand, for \( C^\text{intra}_\alpha [g] \), when \( N \to +\infty \), we have:
\[
\frac{1}{N} \sum_{j \neq i} \alpha_{ij} = \frac{1}{N} \sum_{j \neq i} K(x_0[i], x_0[j]) \to \int K(x_0, y_0)\mathbb{P}(y_0)dy_0
\]
(33)
\[
\sum_j \phi_i(x_0) \phi_j(y_0)\mathbb{P}(y_0)dy_0 = \sum_j z_j \phi_i(x_0)
\]
(34)
Therefore, we have:
\[
C^\text{intra}_\alpha [g] \to \frac{1}{2} \sum_l z_l \int \phi_i(x_0)(g(x) - g(x'))(g(x) - g(x'))\top \mathbb{P}(x, x'|x_0)\mathbb{P}(x_0)dx dx'
\]
(35)
Similarly,
\[
\int (g(x) - g(x'))(g(x) - g(x'))\top \mathbb{P}(x, x'|x_0)dx dx'
\]
(36)
\[
= 2 \int g(x)g(x)\top \mathbb{P}(x|x_0)dx - 2 \int g(x)\mathbb{P}(x|x_0)dx \int g(x')\mathbb{P}(x'|x_0)dx'
\]
(37)
\[
= 2 \mathbb{E}_{x \sim \text{aug}(\cdot|x_0)}[gg^\top] - 2 \mathbb{E}_{x \sim \text{aug}(\cdot|x_0)}[g]\mathbb{E}_{x \sim \text{aug}(\cdot|x_0)}[g^\top]
\]
(38)
\[
= 2\mathbb{V}_{x \sim \text{aug}(\cdot|x_0)}[g]
\]
(39)
So we have:
\[
C^\text{intra}_\alpha [g] \to \frac{1}{2} \sum_l z_l \int \phi_i(x_0)2\mathbb{V}_{x \sim \text{aug}(\cdot|x_0)}[g]\mathbb{P}(x_0)dx_0
\]
(40)
\[
= \sum_l z_l^2 \mathbb{E}_{x_0 \sim \hat{p}_i} \mathbb{V}_{x \sim \text{aug}(\cdot|x_0)}[g]
\]
(41)
Using the law of total variation, finally we have:
\[
C_\alpha [g] \to \sum_l z_l^2 \mathbb{V}_{x_0 \sim \hat{p}_i} \mathbb{E}_{x \sim \text{aug}(\cdot|x_0)}[g]
\]
(42)
B ONE-LAYER MODEL (SEC. 3)

B.1 COMPUTATION OF THE TWO EXAMPLE MODELS

Here we assume ReLU activation $h(x) := \max(x, 0)$, which is a homogeneous activation $h(x) = h'(0)x$. Note that we consider $h'(0) = 0$. Therefore, for any sample $x$, if $w \cdot x = 0$, then we don't consider it to be included in the active region of ReLU, i.e., $x^w = x \cdot h'(w^T x) = 0$.

Let $z$ be a hidden binary variable and we could compute $A(w)$ (here $p_0 := \mathbb{P}[z = 0]$ and $p_1 := \mathbb{P}[z = 1]$):

$$\nabla[\tilde{x}^w] = \nabla_z[\mathbb{E}[\tilde{x}^w|z]] + \nabla_z[\mathbb{V}[\tilde{x}^w|z]] = p_0 p_1 \Delta(w) \Delta^T(w) + p_0 \Sigma_0(w) + p_1 \Sigma_1(w) \quad (43)$$

where $\Delta(w) := \mathbb{E}[\tilde{x}|z = 1] - \mathbb{E}[\tilde{x}|z = 0]$ and $\Sigma_z(w) := \mathbb{V}[\tilde{x}|z]$.

Latent categorical model. If $w = u_m$, let $z := \mathcal{I}(y = m)$. This leads to $\Sigma_1(u_m) = \Sigma_0(u_m) = 0$ and $\Delta(u_m) = u_m$. Therefore, we have:

$$A(w)|_{w = u_m} := \mathcal{C}_\alpha[\tilde{x}^w] = \mathbb{V}[\tilde{x}^w] = \mathbb{P}[y = m] (1 - \mathbb{P}[y = m]) u_m u_m^T \quad (44)$$

Latent summation model. If $w = u_m$, first notice that due to orthogonal constraints we have $w^T x = \sum_m y_m u_m^T w = y_m$. Let $z := \mathcal{I}(y_m > 0)$, then we can compute $\Delta(u_m) = y_m u_m, \Sigma_1(u_m) = I - u_m u_m^T$ and $\Sigma_0(u_m) = 0$. Therefore, we have:

$$A(w)|_{w = u_m} := \mathcal{C}_\alpha[\tilde{x}^w] = \mathbb{V}[\tilde{x}] = (1 - q_m)^2 u_m u_m^T + q_m (I - u_m u_m^T) \quad (45)$$

B.2 DERIVATION OF TRAINING DYNAMICS

Lemma 3 (Training dynamics of 1-layer network with homogeneous activation in contrastive learning). The gradient dynamics of Eqn. (5) is (note that $\alpha$ is treated as an independent variable):

$$\dot{w}_k = P_{w_k} A(w_k) w_k \quad (7)$$

Here $P_{w_k} := I - w_k w_k^T$ projects a vector into the complementary subspace spanned by $w_k$.

Proof. First of all, it is clear that from Eqn. (5) each $w_k$ evolves independently. Therefore, we omit the subscript $k$ and derive the dynamics of one node $w$.

To compute the training dynamics, we only need to compute the differential of $\mathcal{C}_\alpha[h(w^T x)]$. We use matrix differential form [Giles, 2008] to make the derivation easier to understand.

Note that for one-layer network with $K = 1$ nodes, $E(w) := \frac{1}{2} \mathcal{C}_\alpha[h(w^T x)] = \frac{1}{2} \mathcal{C}_\alpha[h(w^T x), h(w^T x)]$ be the objective function to be maximized. Using the fact that

- $\mathcal{C}_\alpha[x, y]$ is a bilinear form (linear w.r.t $x$ and $y$) given fixed $\alpha$,
- for any vector $a$ and $b$, we have $a^T \mathcal{C}_\alpha[x, y] b = \mathcal{C}_\alpha[a^T x, b^T y]$,
- for scalar $x$ and $y$, $\mathcal{C}_\alpha[x, y] = \mathcal{C}_\alpha[y, x]$,

and by the product rule $d(x \cdot y) = dx \cdot y + x \cdot dy$, we have:

$$dE = \frac{1}{2} \mathcal{C}_\alpha[h(w^T x), h'(w^T x) dw^T x] + \frac{1}{2} \mathcal{C}_\alpha[h'(w^T x) dw^T x, h(w^T x)]$$

$$= \mathcal{C}_\alpha[h(w^T x), h'(w^T x) dw]$$

(46)

Now use the homogeneous condition (Assumption [a]) for activation $h$: $h(x) = h'(x)x$, which gives $h(w^T x) = h'(w^T x) w^T x$, therefore, we have:

$$dE = w^T \mathcal{C}_\alpha[h'(w^T x) x, h'(w^T x) x] dw = w^T A(w) dw$$

(47)

where $A(w) := \mathcal{C}_\alpha[h'(w^T x) x, h'(w^T x) x] = \mathcal{C}_\alpha[\tilde{x}^w, \tilde{x}^w]$. Therefore, by checking the coefficient associated with the differential form $dw$, we know $\frac{\partial w}{\partial \theta} = A(w) w$. By gradient ascent, we have $\dot{w} = A(w) w$. Since $w$ has the additional constraint $||w||_2 = 1$, the final dynamics is $\dot{w} = P_{w} A(w) w$ where $P_{w} := I - w w^T$ is a projection matrix that projects a vector into the orthogonal complement subspace of the subspace spanned by $w$. □


**Remarks.** Note that an alternative route is to use homogeneous condition first: \( C_α[h(w^T x)] = w^T A(x)w \), then taking the differential. This involves an additional term \( \frac{1}{2}w^T (dA)w \). In the following we will show it is zero. For this we first compute \( dA \):

\[
dA = dC_α[h'(w^T x)] = C_α[h''(w^T x)h(w^T x)x] + C_α[h'(w^T x)h''(w^T x)(dw^T x)]
\]

Therefore, since \( a^T C_α[a, y]b = C_α[a^T x, b^T y] \), we have:

\[
w^T (dA)w = C_α[(dw^T x)h''(w^T x)w^T x, h(w^T x)x] + C_α[h(w^T x), h''(w^T x)(dw^T x)w^T x]
\]

\[
= 2C_α(A(w)w^T x, h(w^T x))
\]

Note that we now see the term \( h''(w^T x)w^T x \). For ReLU activation, its second derivative \( h''(x) = \delta(x) \), where \( \delta(x) \) is Direct delta function \([\text{Boas & Peters}, 1984]\). From the property of delta function, we have \( x\delta''(x) = x\delta'(x) = 0 \) even evaluated at \( x = 0 \). Therefore, \( h''(w^T x)w^T x = 0 \) and \( w^T (dA)w = 0 \). This is similar for LeakyReLU as well.

**B.3 Local Stability**

**Theorem 1** (Stability of \( w^* \)). If \( w_* \) is a LME of \( A(w_*) \) and \( \lambda_{\text{gap}}(w_*) > \rho(w_*) \), then \( w_* \) is stable.

**Proof.** For any unit direction \( \|u\|_2 = 1 \) so that \( u^T w_* = 0 \), consider the perturbation \( v = \sqrt{1 - \epsilon^2}w_* + \epsilon u \). Since \( \|w_*\|_2 = 1 \) we have \( \|v\|_2 = 1 \).

Now let's compute \( P_w^\perp A(v)v \). First, we have:

\[
P_w^\perp = I - vv^T = I - \left(\sqrt{1 - \epsilon^2}w_* + \epsilon u\right)\left(\sqrt{1 - \epsilon^2}w_* + \epsilon u\right)^T
\]

\[
= I - w_*w_*^T - \epsilon(uw_*^T + w_*u^T) + O(\epsilon^2)
\]

\[
= P_w^\perp - \epsilon(uw_*^T + w_*u^T) + O(\epsilon^2)
\]

So we have:

\[
P_w^\perp A(v)v = P_w^\perp A(w_*)v - \epsilon(uw_*^T + w_*u^T)A(w_*)v + O(\epsilon^2)
\]

\[
= P_w^\perp A(w_*)v - \epsilon\lambda_*u + O(\epsilon^2)
\]

The previous derivation is due to the fact that \( P_w^\perp A(w_*)w_* = 0, u^T A(w_*)w_* = 0 \) and \( P_w^\perp u = u \). Therefore, for \( P_w^\perp A(v)v \), we can decompose it to two parts:

\[
P_w^\perp A(v)v = P_w^\perp A(w_*)v + P_w^\perp (A(v) - A(w_*))v
\]

\[
= P_w^\perp (A(w_*) - \lambda_*I)u + P_w^\perp (A(v) - A(w_*))v + O(\epsilon^2)
\]

Therefore, since \( u^T w_* = 0 \), we have:

\[
u^T P_w^\perp (A(w_*) - \lambda_*I)u = u^T (I - w_*w_*^T)(A(w_*) - \lambda_*I)u
\]

\[
= \epsilon u^T (A(w_*) - \lambda_*I)u \leq -\lambda_{\text{gap}}(w_*)\epsilon + O(\epsilon^2)
\]

and since \( \|u\|_2 = \|v\|_2 = 1 \) and \( \|P_w^\perp\|_2 = 1 \), we have:

\[
|u^T P_w^\perp (A(v) - A(w_*))v| \leq ||(A(v) - A(w_*))v||_2
\]

By the definition of local roughness measure \( \rho(w_*) \), we have:

\[
||A(v) - A(w_*)||_2 \leq \rho(w_*)||v - w_*||_2 + O(||v - w_*||_2^2) = \rho(w_*)\epsilon + O(\epsilon^2)
\]

This leads to

\[
||A(v) - A(w_*)||_2 \leq \rho(w_*)\epsilon + O(\epsilon^2)
\]

Therefore, we have:

\[
u^T P_w^\perp A(v)v \leq - (\lambda_{\text{gap}}(w_*) - \rho(w_*))\epsilon + O(\epsilon^2)
\]

When \( \lambda_{\text{gap}}(w_*) > \rho(w_*) \) and we have \( u^T P_w^\perp A(v)v < 0 \) for any \( u \perp w_* \) and sufficiently small \( \epsilon \). Therefore, the critical point \( w_* \) is stable. □
Theorem 2 (Bound of local roughness $\rho(w)$ in ReLU setting). If input $\|x\|_2 \leq C_0$ is bounded, $\alpha$ has kernel structure (Def. 7) and batchsize $N \to +\infty$, then $\rho(w_*) \leq \frac{C_2^2 \nu(C_0)}{\pi} r(w_*, \alpha)$, where $r(w, \alpha) := \sum_{i=0}^{+\infty} \frac{\gamma^2(\alpha)}{\gamma^2_1} \max_{x_0 \in \Omega^i} \hat{p}_i(x; \alpha)$.

Proof. Suppose $w_*$ and its local perturbation $w$ are on the unit sphere $\|w\|_2 = \|w_*\|_2 = 1$. Since $w$ is a local perturbation, we have $w^T w_* \geq 1 - \epsilon$ for $\epsilon \ll 1$.

In the following we will check how we bound $\|(A(w) - A(w_*))w_*\|_2$ in terms of $\|w - w_*\|_2$ and then we can get the upper bound of local roughness metric $\rho(w_*)$.

Let the function $g(x) := \tilde{x}^w$, apply Corollary 1 with no augmentation and the large batch limits, we have

$$A(w) := C_\alpha \tilde{x}^w = \sum_l z_l^2 \tilde{V}_{\tilde{p}_l} \tilde{x}^w,$$

where $\tilde{p}_l(x) = \frac{1}{z_l^2} p_l(x) \phi_l(x)$ is the probability distribution of the input $x$, adjusted by the mapping of the kernel function determined by the pairwise importance $\alpha_{ij}$ (Def. 1, 2) is its normalization constant.

To study $(A(w) - A(w_*))w_*$, we will study each component $(\tilde{V}_{\tilde{p}_l} \tilde{x}^w) w_*$.

Note that since $\tilde{x}^w := x([w^T x \geq 0], we have $V_{\tilde{p}_l} \tilde{x}^w = E_{\tilde{p}_l} \tilde{x}^w \|w^T x \geq 0\| - E_{\tilde{p}_l} \tilde{x}^w \|w^T x \geq 0\|)$. Let

$$e := \int_{w^T x \geq 0} x \tilde{p}_l(x) dx, \quad e_* := \int_{w_*^T x \geq 0} x \tilde{p}_l(x) dx$$

$$E := \int_{w^T x \geq 0} xx^T \tilde{p}_l(x) dx, \quad E_* := \int_{w_*^T x \geq 0} xx^T \tilde{p}_l(x) dx$$

So we can write

$$\tilde{V}_{\tilde{p}_l} \tilde{x}^w = E - ee^T, \quad \tilde{V}_{\tilde{p}_l} \tilde{x}^w = E_* - ee_*^T$$

and $\tilde{V}_{\tilde{p}_l} \tilde{x}^w - \tilde{V}_{\tilde{p}_l} \tilde{x}^{w_*} = (E - E_*) + (e_*e_*^T - ee^T)$.

Define the following regions

$$\Omega_+ := \{x : w_*^T x \geq 0, w^T x \leq 0\}$$

$$\Omega_- := \{x : w_*^T x \leq 0, w^T x \geq 0\}$$

$$\Omega := \Omega_+ \cup \Omega_-$$

Now let’s bound $(E - E_*)w_*$ and $(e_*e_*^T - ee^T)w_*$.

**Bound $(E - E_*)w_*$.** We have:

$$E - E_* = \int_{\Omega_-} xx^T \tilde{p}_l(x) dx - \int_{\Omega_+} xx^T \tilde{p}_l(x) dx$$

and thus

$$(E - E_*)w_* = \int_{\Omega_-} xx^T w_* \tilde{p}_l(x) dx - \int_{\Omega_+} xx^T w_* \tilde{p}_l(x) dx$$

For any $x \in \Omega_+$, we have:

$$0 \leq w_*^T x = w^T x + (w_* - w)^T x \leq (w_* - w)^T x \leq C_0 \|w_* - w\|_2$$

Therefore, $|w_*^T x| \leq M \|w_* - w\|_2$ and we have

$$\left\| \int_{\Omega_+} xx^T w_* \tilde{p}_l(x) dx \right\|_2 \leq \int_{\Omega_+} |w_*^T x| \|x\|_2 \tilde{p}_l(x) dx$$

$$\leq C_0^2 \|w_* - w\|_2 \max_{x \in \Omega_+} \tilde{p}_l(x) \int_{\Omega_+} \|x\|_2 \leq C_0 \frac{\nu(C_0)}{2\pi} \arccos w^T w_*$$
where $\text{vol}(C_0)$ is the volume of the $d$-dimensional ball of radius $C_0$. Similarly for $x \in \Omega_-$, we have
\[
0 \geq w^T x = w^T x + (w_\ast - w)^T x \geq (w_\ast - w)^T x \geq -C_0 \|w_\ast - w\|_2
\]
and overall we have:
\[
\|(E - E_\ast)w_\ast\|_2 \leq \frac{C_0^3 \text{vol}(C_0)}{\pi} \|w_\ast - w\|_2 \max_{x \in \Omega} \tilde{p}_l(x) \arccos w^T w_\ast
\]
(79)
Since for $x \in (0, 1]$, $\arcsin \sqrt{1 - x^2} \leq \sqrt{1 - (w^T w_\ast)^2}$, we have:
\[
\arcsin w^T w_\ast = \arcsin \sqrt{1 - (w^T w_\ast)^2} \leq \frac{\sqrt{2(1 - w^T w_\ast)}}{w^T w_\ast}
\]
\[
= \frac{1}{1 - \epsilon} \|w - w_\ast\|_2
\]
we have:
\[
\|(E - E_\ast)w_\ast\|_2 \leq \frac{C_0^3 \text{vol}(C_0)}{\pi} \frac{1}{1 - \epsilon} \|w_\ast - w\|_2 \max_{x \in \Omega} \tilde{p}_l(x)
\]
(80)
Therefore, $\|(E - E_\ast)w_\ast\|_2$ is a second-order term w.r.t. $\|w - w_\ast\|_2$.

**Bound** $(e_\ast e_\ast^T - ee^T)w_\ast$. On the other hand:
\[
ee^T - e_\ast e_\ast^T = e(e - e_\ast)^T + (e - e_\ast)e_\ast^T
\]
We have $\|e\|_2, \|e_\ast\|_2$ bounded and
\[
e - e_\ast = \int_{\Omega_-} x\tilde{p}_l(x)dx - \int_{\Omega_+} x\tilde{p}_l(x)dx
\]
(81)
Using similar derivation, we conclude that $\|e(e - e_\ast)^T w_\ast\|_2$ is also a second-order term. The only first-order term is $\|(e - e_\ast)e_\ast^T w_\ast\|_2$:
\[
\|(e - e_\ast)e_\ast^T w_\ast\|_2 \leq \text{E}_l[|h(w^T x)|] \|e\|_2 \|\tilde{p}_l(x)\|dx
\]
(82)
\[
\leq C_0^2 \int_{\Omega} \tilde{p}_l(x)dx \leq C_0^2 \max_{x \in \Omega} \tilde{p}_l(x) \int_{\Omega: \|x\|_2 \leq C_0} dx
\]
(83)
\[
\leq \frac{C_0^3 \text{vol}(C_0)}{\pi} \arcsin w^T w_\ast \max_{x \in \Omega} \tilde{p}_l(x)
\]
\[
\leq \frac{C_0^3 \text{vol}(C_0)}{\pi} \frac{1}{1 - \epsilon} \|w - w_\ast\|_2 \max_{x \in \Omega} \tilde{p}_l(x)
\]
Overall we have:
\[
\|(A(w) - A(w_\ast))w_\ast\|_2 \leq \sum_{i} z_i^2 \|\nabla \tilde{p}_l[\hat{x}^m] - \nabla \tilde{p}_l[\hat{x}^0]\|w_\ast\|_2
\]
(84)
\[
\leq \frac{C_0^3 \text{vol}(C_0)}{\pi} \frac{1}{1 - \epsilon} \left( \sum_{i} z_i^2 \max_{x \in \Omega} \tilde{p}_l(x) \right) \|w - w_\ast\|_2 + O(\|w - w_\ast\|_2^2)
\]
(85)
Since $\rho(w_\ast)$ is the smallest scalar that makes the local roughness metric hold and $\epsilon$ is arbitrarily small, we have:
\[
\rho(w_\ast) \leq \frac{C_0^3 \text{vol}(C_0)}{\pi} r(w_\ast, \alpha)
\]
(86)
where $r(w, \alpha) := \sum_{i} z_i^2 \max_{x = 0} \tilde{p}_l(x; \alpha)$.

**Corollary 2** (Effect of different $\alpha$). For uniform $\alpha_\ast (\alpha_{ij} := 1)$ and 1-D Gaussian $\alpha_\delta (\alpha_{ij} := \exp(-\|h(w^T x_0[i]) - h(w^T x_0[j])\|^2/2\tau))$, we have $r(w_\ast, \alpha_\ast) = z_0(\alpha_\ast)r(w_\ast, \alpha_\ast)$ with $z_0(\alpha_\ast) := \int \exp(-h^2(w_\ast^T x)/2\tau) p_\Omega(x)dx \leq 1$. As a result, $z_0(\alpha_\ast) \ll 1$ leads to $r(w_\ast, \alpha_\ast) \ll r(w_\ast, \alpha_\ast)$. 

\[\square\]
Then Power Iteration (Eqn. PI) converges to a critical point. Along the trajectory, let

\begin{align*}
\text{(Existence of critical points)}
\end{align*}

Theorem 3

Therefore, we have:

\begin{align*}
\text{Here}
\end{align*}

For Gaussian \(\alpha_g\), from Lemma [1] we know that its infinite-dimensional mapping \(\phi_g(x)\) has the following form for \(w = w_*\):

\begin{align*}
\phi_g(x) = e^{-\frac{\kappa^2 x^T x}{2\sigma^2}} \begin{bmatrix} 1 \\ \frac{\tau^{-1/2} h(w^T x)}{\tau^{1/2} \sqrt{2\pi}} h^2(w^T x) \\ \vdots \\ \frac{\tau^{-1/2} h^k(w^T x)}{\tau^{1/2} \sqrt{2\pi}} \end{bmatrix}
\end{align*}

When \(l \geq 1\), \(z(l) = z(t) \phi_0(x)P_D(x) = 0\) for any \(x\) on the plane \(w^T x = 0\), since \(\phi_0(x) = 0\) on the plane. On the other hand, \(\phi_0(x) = e^{-\frac{\kappa^2 x^T x}{2\sigma^2}}\). On the plane, \(\phi_0(x) = 1\) and is a constant. Therefore, we have:

\begin{align*}
r(w_*, \alpha_g) := \sum_{l=0}^{\infty} z(l) = \max_{u, x=0} \phi_0(x) = \max_{u, x=0} \phi_0(x) = 0 \text{ for } w = w_*, (98)\n\end{align*}

\begin{align*}
r(w_*, \alpha_g) := \sum_{l=0}^{\infty} z(l) = \max_{u, x=0} \phi_0(x)P_D(x) = 0 \text{ for } w = w_*, (99)\n\end{align*}

Here

\begin{align*}
z_0(\alpha_g) := \int \phi_0(x)P_D(x)dx = \int e^{-\frac{\kappa^2 x^T x}{2\sigma^2}} P_D(x)dx \leq 1 \quad (100)\n\end{align*}

\section{Finding critical points with initial guess (Sec. 3.3)}

\textbf{Notation.} Let \(\lambda_i(w)\) and \(\phi_i(w)\) be the \(i\)-th eigenvalue and unit eigenvector of \(A(w)\) where \(\phi_1(w)\) is the largest. We first assume \(A(w)\) is positive definite (PD) and then remove this assumption later. In this case, \(\lambda_1(w) \geq \lambda_2(w) \geq \ldots \geq \lambda_d(w) > 0\). Let \(c(w) := w^T \phi_1(w)\) be the inner product between \(w\) and the maximal eigenvector of \(A(w)\).

Consider the following Power Iteration (PI) format:

\begin{align*}
\hat{w}(t+1) \leftarrow A(w(t))w(t), \quad w(t+1) \leftarrow \frac{\hat{w}(t+1)}{\|\hat{w}(t+1)\|_2} \quad (101)
\end{align*}

Along the trajectory, let \(\phi_i(t) := \phi_i(A(w(t)))\) be the \(i\)-th unit eigenvector of \(A(w(t))\) and \(\lambda_i(t)\) to be the \(i\)-th eigenvalue. Define \(\delta(t) := w(t+1) - w(t)\), \(\delta A(t) := A(w(t+1)) - A(w(t))\), and

\begin{align*}
c_t := c(w(t)) = \phi_1^T(t)w(t), \quad d_t := \phi_1^T(t)w(t+1) \quad (102)
\end{align*}

Then \(-1 \leq c_t, d_t \leq 1\) since they are inner products of two unit vectors.

\textbf{Theorem 3} (Existence of critical points). Let \(c_0 := c(\phi(0)) \neq 0\). If there exists \(\gamma < 1\) so that:

\begin{align*}
\sup_{w \in \mathcal{B}_\gamma} \omega(w) \leq \gamma, \quad (11)
\end{align*}

where \(\mathcal{B}_\gamma := \left\{ w : \omega(w) \geq \frac{c_0 - c_{\gamma}}{1 - \frac{c_0}{c_{\gamma}}}, \quad c_{\gamma} := \frac{2\gamma}{1 + \gamma} \right\}\) is the neighborhood of initial value \(w(0)\). Then Power Iteration (Eqn. PI) converges to a critical point \(w_* \in \mathcal{B}_\gamma\) of Eqn. [7].
Proof. Note that if \( c_0 < 0 \), we can always use \(-\phi_1(w)\) as the maximal eigenvector.

First we assume \( A(w) \) is positive definite (PD) over the entire unit sphere \( \|w\|_2 = 1 \), then follow Lemma 11 and notice that \( \|w - w(0)\|_2 = \sqrt{2(1 - w^\top w(0))} \), so

\[
\|w - w(0)\|_2 \leq \frac{\sqrt{2(1 + \gamma)(1 - c_0)}}{1 - \sqrt{\gamma}} \iff w^\top w(0) \geq \frac{c_0 - c_\gamma}{1 - c_\gamma} \tag{103}
\]

When \( A(w) \) is not PD, Theorem 3 still applies to the PD matrix \( \hat{A}(w) := A(w) - \lambda_{\min}(w)I + \epsilon I \) with \( \lambda \) and \( \kappa \) specified by \( \hat{A}(w) \), where \( \epsilon > 0 \) is a small constant.

This transformation keeps \( c_0 \) since the eigenvectors of \( \hat{A}(w) \) are the same as \( A(w) \). The resulting fixed point \( \hat{w}_* \) is also the fixed point of the original problem with \( A(w) \), due to the fact that

\[
P_w^\perp \hat{A}(w)w = P_w^\perp A(w)w - (\lambda_{\min}(w) - \epsilon)P_w^\perp w = P_w^\perp A(w)w \tag{104}
\]

Remarks. Note that Lemma 11 assumes that along the trajectory \( \{w(t)\} \), \( \mu_t + \nu_t \leq \gamma \) holds. In Theorem 3, this can not be assumed true until we prove that the entire trajectory is within \( B_\gamma \).

B.5 The Effect of Data Augmentation on Local Optima

While the majority of the analysis focuses on the cases where there are no data augmentation (i.e., using Corollary 1), the original formulation Lemma 2 can still handle contrastive learning in the presence of data augmentation.

In fact, data augmentation plays an important role by removing unnecessary local optima. First, Lemma 2 tells that the objective Eqn. 5, when \( K = 1 \), takes the following form:

\[
2\mathcal{E}_\alpha(w) := C_\alpha \left[ h(w^\top x) \right] \rightarrow \sum_l \mathbb{E}_{x_0 \sim \bar{R}_l(\cdot)c_\alpha} [b(w|x_0)] \tag{105}
\]

where \( b(w|x_0) := \mathbb{E}_{x \sim p_{aug}(\cdot|x_0)} [h(w^\top x)|x_0] \).

Now let us consider the following simple data augmentation of \( x_0 \):

\[
x = R(t)x_0, \quad t \sim \text{Uniform}(\mathcal{T}) \tag{106}
\]

where \( R(t) \in \mathbb{R}^{d \times d} \) is some rotation parameterized by \( t \), which is drawn uniformly from a parameter family \( \mathcal{T} \).

We assume \( \{R(t)\}_{t \in \mathcal{T}} \) forms a 1-dimensional Lie group parameterized by \( \mathcal{T} \). This means that

- **Closeness.** For any \( t, t' \in \mathcal{T} \), there exists \( t'' \in \mathcal{T} \) so that \( R(t'') = R(t)R(t') \).
- **Existence of inverse element.** For each \( R(t) \), there exists an inverse element \( t' \in \mathcal{T} \) so that \( R(t') = R^{-1}(t) = R^\top(t) \). The last equality is due to the fact that \( R(t) \) is a rotation.
- **Existence of identity map.** \( R(0) = I \).

Then for any small transformation \( R(t') \) applied to the weights \( w \) (here “small” means \( \|R(t') - I\|_2 \) is small), we can write down \( b(R(t')w|x_0) \) using reparameterization trick:

\[
b(R(t)w|x_0) := \mathbb{E}_{x \sim p_{aug}(\cdot|x_0)} [h((R(t)w)^\top x)|x_0] = \int h(w^\top R^\top(t)R(t')x_0)|\mathbb{P}[t']|dt' = \int h(w^\top R^{-1}(t)R(t')x_0)|\mathbb{P}[t']|dt' = h(w^\top R(t'')x_0)|\mathbb{P}[t'']|dt'' = b(w|x_0) \tag{107}
\]

Note that the last equality is due to the fact that \( \{R(t)\}_{t \in \mathcal{T}} \) is a Lie group, so that \( R^{-1}(t)R(t') \) always maps to another group element \( R(t'') \), and \( t'' \) as the resulting parameterization is still uniform.
Due to stop gradient, $\alpha$ and thus $\phi_l(\cdot; \alpha)$ is treated as a constant term when checking the local property of the current parameters $w$. This means that in the local neighborhood of $w$, $E_\alpha(w) = E_\alpha(R(t')w)$.

Now notice an important observation: if $w' := R(t')w \neq w$, then $E_\alpha(w) = E_\alpha(R(t')w) = E_\alpha(w')$ and therefore, $w$ cannot be a local optimal.

Intuitively, this means that the data augmentation can remove certain local optima of $w$, if they are not **locally invariant** (i.e., $R(t')w \neq w$) to the transformation of the data augmentation. Therefore, augmentation removes certain patterns in the input data and their local optima in the training, to only keep patterns (local optima) that are most relevant to the tasks.

Here we only use 1-dimensional rotation group as one simple example. In practice, the augmentation may not globally form a Lie group, and there could be multiple different types of augmentations, yielding high-dimensional transformation space. Therefore, we may use Lie algebra instead to capture the local transformation structure, without making assumptions about the global structure. We will give a formal study in the future work.

C TWO LAYER CASE (SEC. 4)

C.1 LEARNING DYNAMICS

**Lemma 4** (Dynamics of 2-layer nonlinear network with contrastive loss).

$$
\dot{V} = VC_\alpha[f_1], \quad \dot{w} = P_w^+ [(S \otimes 1_d 1_d^\top) \circ C_\alpha[\tilde{x}]] w
$$

where $1_d$ is $d$-dimensional all-one vector, $\otimes$ is Kronecker product and $\circ$ is Hadamard product.

**Proof.** The output of the 2-layer network can be written as the following:

$$
f_{2l} = \sum_k v_{ik} h(w_k^\top x_k)
$$

For convenience, we use $f_1 := [h(w_k^\top x_k)]$ to represent the column vector that collects all the outputs of intermediate nodes, and $v_1^\top$ is the $l$-th row vector in $V$.

According to Theorem 1 in [Han2022], the gradient descent direction of contrastive loss corresponds to the gradient ascent direction of the energy function $E_\alpha(\theta)$. From Eqn. 25 of that theorem, we have:

$$
\frac{\partial E}{\partial \theta} = \sum_l C_\alpha \left[ \frac{\partial f_{2l}}{\partial \theta}, f_{2l} \right]
$$

Therefore, for $V = [v_{ik}]$ we have:

$$
\dot{v}_i = \frac{\partial E}{\partial v_i} = \sum_l C_\alpha \left[ \frac{\partial f_{2l}}{\partial v_i}, f_{2l} \right] = C_\alpha \left[ f_1, v_i^\top f_1 \right] = C_\alpha \left[ f_1, f_1 \right] v_i
$$

So we have $\dot{v}_i = C_\alpha[f_1]v_i$, or $\dot{V} = VC_\alpha[f_1]$. 

22
Now we compute $\partial E / \partial w_k$:

$$ w_k = \frac{\partial E}{\partial w_k} = \sum_l C_{\alpha} \left[ \frac{\partial f_{2l}}{\partial w_k}, f_{2l} \right] $$

$$ = \sum_l C_{\alpha} [v_{lk}' (w_k^T x_k), v_l^T f_1] $$

$$ = \sum_l v_{lk} C_{\alpha} [x_k, v_l^T f_1] $$

$$ = \sum_l v_{lk} C_{\alpha} [\tilde{x}_k, \sum_{k'} v_{lk'} h(w_{k'}^T x_k')] $$

$$ = \sum_{k'} \left( \sum_l v_{lk} v_{lk'} \right) C_{\alpha} [\tilde{x}_k, \tilde{x}_{k'}] w_{k'} $$

$$ = \sum_{k'} s_{kk'} C_{\alpha} [\tilde{x}_k, \tilde{x}_{k'}] w_{k'} $$

where $S = [s_{kk'}] = V^T V = \sum_l v_l v_l^T$. Let $w := [w_1; \ldots; w_K]$ and it leads to the conclusion. When $M > 1$, the proof is similar.

\[\square\]

### C.2 Variance Decomposition

Let $p_c := P[z = c]$ be the probability that the latent variable $z$ takes categorical value $c$.

**Lemma 5** (Close-form of variance under Assumption 2). With Assumption 2 we have

$$ \mathbb{V}[\tilde{x}] = \mathbb{E}_z \mathbb{V}[\tilde{x}|z] + \mathbb{V}_z \mathbb{E}[\tilde{x}|z] $$

where $L_k := \mathbb{E}_z \mathbb{V}[\tilde{x}_k]|z \in \mathbb{R}^{M_d}$ and $\Delta(c) := \mathbb{E}[\tilde{x}|z = c] - \mathbb{E}[\tilde{x}|z \neq c] \in \mathbb{R}^{MK_d}$. In particular when $C = 2$, the second term becomes $p_0p_1 \Delta \Delta^T$, a rank-1 matrix. Here $\Delta := \Delta(0)$ for brevity.

**Proof.** Use variance decomposition, we have:

$$ \mathbb{V}[\tilde{x}] = \mathbb{E}_z \mathbb{V}[\tilde{x}|z] + \mathbb{V}_z \mathbb{E}[\tilde{x}|z] $$

Remember that $\tilde{x}_{km}$ is an abbreviation of gated input:

$$ \tilde{x}_{km} := \tilde{x}_k^{wk_m} := x_k \cdot h'(w_{km}^T x_k) $$

By conditional independence, we have

$$ \text{Cov}[\tilde{x}_{km}, \tilde{x}_{k'm'}|z] = 0 \quad \forall k \neq k' $$

This is because $\tilde{x}_{km}$ and $\tilde{x}_{k'm'}$ are deterministic functions of $x_k$ and $x_{k'}$ and thus are also independent of each other.
Let
\[
\tilde{x}_k := \begin{bmatrix} \tilde{x}_{k1} \\ \tilde{x}_{k2} \\ \vdots \\ \tilde{x}_{kM} \end{bmatrix} \in \mathbb{R}^{Md}
\] (124)

and \( L_k := \mathbb{E}_z V[\tilde{x}_k|z] \). Then we know that \( \mathbb{E}_z V[\tilde{x}|z] = \text{diag}_k[L_k] \) is a block diagonal matrix (See Fig. 6).

On the other hand, \( \mathbb{V}_z \mathbb{E}[\tilde{x}|z] \) is a low-rank matrix:
\[
\mathbb{V}_z \mathbb{E}[\tilde{x}|z] = \mathbb{E}_z \left[ \left( \mathbb{E}[\tilde{x}|z] - \mathbb{E}[\tilde{x}] \right) \left( \mathbb{E}[\tilde{x}|z] - \mathbb{E}[\tilde{x}] \right)^\top \right] 
\] (125)

Let \( q_c := \mathbb{E}[\tilde{x}|z = c] \) and \( q_{c-} := \mathbb{E}[\tilde{x}|z \neq c] \), then we have:
\[
\mathbb{E}[\tilde{x}|z = c] - \mathbb{E}[\tilde{x}] = q_c - \sum_c p_c q_c = (1 - p_c) \left( q_c - \sum_{c' \neq c} p_{c'} q_{c'} \right) 
\] (126)
\[
= (1 - p_c) \left( q_c - \sum_{c' \neq c} \mathbb{P}[z = c'|z \neq c] q_{c'} \right) 
\] (127)
\[
= (1 - p_c)(q_c - q_{c-}) 
\] (128)

Therefore, we have:
\[
\mathbb{V}_z \mathbb{E}[\tilde{x}|z] = \mathbb{E}_z \left[ \left( \mathbb{E}[\tilde{x}|z] - \mathbb{E}[\tilde{x}] \right) \left( \mathbb{E}[\tilde{x}|z] - \mathbb{E}[\tilde{x}] \right)^\top \right] 
\] (129)
\[
= \sum_c p_c(1 - p_c)^2 (q_c - q_{c-})(q_c - q_{c-})^\top 
\] (130)
\[
= \sum_c p_c(1 - p_c)^2 \Delta(c) \Delta(c)^\top 
\] (131)

where
\[
\Delta(c) := \Delta(c; W) := q_c - q_{c-} = \begin{bmatrix} \Delta_{11}(c) \\ \cdots \\ \Delta_{KM}(c) \end{bmatrix} \in \mathbb{R}^{KMd} 
\] (132)

and
\[
\Delta_{km}(c) := \Delta_{km}(c; w_{km}) := \mathbb{E}[\tilde{x}_{km}|z = c] - \mathbb{E}[\tilde{x}_{km}|z \neq c] 
\] (133)

We can see that \( \mathbb{V}_z \mathbb{E}[\tilde{x}|z] \) is at most rank-\( C \), since it is a summation of \( C \) rank-1 matrices.

In particular, when \( C = 2 \), it is clear that \( \Delta(0) = -\Delta(1) \) and thus \( \Delta(0) \Delta(0)^\top = \Delta(1) \Delta(1)^\top \) and \( \sum_c p_c(1 - p_c)^2 = p_0 p_1^2 + p_1 p_0^2 = p_0 p_1 \). Hence the conclusion. \( \square \)

### C.3 Global modulation when \( C = 2 \) and \( M = 1 \)

**Theorem 5** (Dynamics of \( w_k \) under conditional independence). When \( C = 2 \) and \( M = 1 \), the dynamics of \( w_k \) is given by (\( s_k^2 \) and \( \delta_k \geq 0 \) are scalars defined in the proof):
\[
w_k = P_{\hat{w}_k} \left( s_k^2 A_k(w_k) + \delta_k \Delta_k \Delta_k^\top \right) w_k 
\] (15)

**Proof.** Since \( M = 1 \), each receptive field (RF) \( R_k \) only output a single node with output \( f_k \). Let:
\[
L_k := \mathbb{E}_z V[\tilde{x}_k|z] 
\] (134)
\[
d_k := w_k^\top L_k w_k = \mathbb{E}_z V[f_k|z] \geq 0 
\] (135)
\[
D := \text{diag}_k[d_k] 
\] (136)
\[
b := |b_k| := |w_k^\top \Delta_k| \in \mathbb{R}^K 
\] (137)

and \( \lambda \) be the maximal eigenvalue of \( \mathbb{V}[f_1] \). Here \( L_k \) is a PSD matrix and \( D \) is a diagonal matrix. Then
\[
\mathbb{V}[f_1] = D + p_0 p_1 b b^\top 
\] (138)
is a diagonal matrix plus a rank-1 matrix. Since $p_0 p_1 b b^T$ is always PSD, $\lambda = \lambda_{\text{max}}(V[f_1]) \geq \lambda_{\text{max}}(D) = \max_k d_k$. Then using Bunch–Nielsen–Sorensen formula [Bunch et al. 1978], for largest eigenvector $s$, we have:

$$s_k = \frac{1}{Z} \frac{b_k}{d_k - \lambda}$$ (139)

where $\lambda$ is the corresponding largest eigenvalue satisfying $1 + p_0 p_1 \sum_k \frac{b_k^2}{d_k - \lambda} = 0$, and $Z = \sqrt{\sum_k \left(\frac{b_k}{d_k - \lambda}\right)^2}$. Note that the above is well-defined, since if $k^* = \arg \max_k d_k$ and $b_{k^*} \neq 0$, then $\lambda > \max_k d_k = d_{k^*}$. So $b_k/(d_k - \lambda)$ won’t be infinite.

So we have:

$$\dot{w}_k = \sum_{k'} s_k s_{k'} C_{kk'} [\bar{x}_k, \bar{x}_{k'}] w_{k'}$$ (140)

$$= \sum_{k'} s_k s_{k'} (L_k \mathbb{1}(k = k') + p_0 p_1 \Delta_k \Delta_{k'}^T) w_{k'}$$

$$= s_k^2 \mathbb{V}[\bar{x}_k] w_k + p_0 p_1 s_k \Delta_k \sum_{k' \neq k} s_{k'} \Delta_{k'}^T w_{k'}$$ (141)

$$= s_k^2 \mathbb{V}[\bar{x}_k] w_k + \frac{p_0 p_1 b_k}{Z^2(d_k - \lambda)} \Delta_k \sum_{k' \neq k} \frac{b_{k'}^2}{d_{k'} - \lambda}$$

$$= s_k^2 \mathbb{V}[\bar{x}_k] w_k + \delta_k \Delta_k \Delta_k^T w_k$$

$$= (s_k^2 \mathbb{V}[\bar{x}_k] + \delta_k \Delta_k \Delta_k^T) w_k$$ (142)

where

$$\delta_k := \frac{p_0 p_1}{Z^2(\lambda - d_k)} \sum_{k' \neq k} \frac{b_{k'}^2}{\lambda - d_{k'}}.$$ (143)

Since $\lambda \geq \max_k d_k$, we have $\delta_k \geq 0$ and thus the modulation term is non-negative. Note that since $p_0 p_1 \sum_k \frac{b_k^2}{\lambda - d_k} = 1$, we can also write $\delta_k = 1 - \frac{p_0 p_1 b_k^2}{\lambda - d_k}$. $\square$

**Theorem 6** (Global modulation of attractive basin). *If the structural assumption holds: $A_k(w_k) = \sum_l g(u_l^T w) u_l u_l^T$ with $g(\cdot) > 0$ a linear increasing function and $\{u_l\}$ orthonormal bases, then for $A_k + c u_l u_l^T$, its attractive basin of $w_k = u_l$ is larger than $A_k$’s for $c > 0$.*

**Proof.** Since $A_k(w) = \sum_l g(u_l^T w) u_l u_l^T$, we could write down its dynamics (we omit the projection $P_w^T$ for now):

$$\dot{w} = A_k(w)w = \sum_l g(u_l^T w) u_l u_l^T w$$ (144)

Let $y_l(t) := u_l^T w(t)$, i.e., $y_l(t)$ is the projected component of the weight $w(t)$ onto the $l$-th direction, i.e., a change of bases to orthonormal bases $\{u_l\}$, then the dynamics above can be written as

$$\dot{y}_l = g(y_l)y_l$$ (145)

which is the same for all $l$, so we just need to study $\dot{x} = g(x)x$. $g(x) > 0$ is a linear increasing function, so we can assume $g(x) = ax + b$ with $a > 0$. Without loss of generality, we could just set $a = 1$.

Then we just want to analyze the dynamics:

$$\dot{y}_l = (y_l + b_l)y_l, \quad b_l > 0$$ (146)

which also includes the case of $A_k + c u_l u_l^T$, that basically sets $b_l = b + c$. Solving the dynamics leads to the following close-form solution:

$$\frac{y_l(t)}{y_l(0)} + b_l = \frac{y_l(0)}{y_l(0) + b_l} e^{b_l t}$$ (147)
Figure 7: The one-dimensional dynamics (Eqn. [146]) ($b_l > 0$). There exists a stable critical point $y_l = -b_l$ and one unstable critical point $y_l = 0$. When $y_l(0) > 0$, the dynamics blows up in finite time $t_l^* = \frac{1}{b_l} \ln \left( 1 + \frac{b_l}{y_l(0)} \right)$. 
The 1-d dynamics has an unstable fixed points \( y_l = 0 \) and a stable one \( y_l = -b_l < 0 \). Therefore, when the initial condition \( y_l(0) < 0 \), the dynamics will converge to \( y_l(+\infty) = -b_l \), which is a finite number. On the other hand, when \( y_l(0) > 0 \), the dynamics has finite-time blow-up \cite{Thompson1990Goriely1998}, i.e., there exists a critical time \( t_l^* < +\infty \) so that \( y_l(t_l^*) = +\infty \). See Fig. 7.

Note that this finite time blow-up is not physical, since we don’t take into consideration of normalization \( Z(t) \), which depends on all \( y_l(t) \). The real quality to be considered is \( \hat{y}_l(t) = \frac{y_l(t)}{Z(t)} y_l(t) \). Fortunately, we don’t need to estimate \( Z(t) \) since we are only interested in the ratio:

\[
  r_{l/l'}(t) := \frac{\hat{y}_l(t)}{\hat{y}_{l'}(t)} = \frac{y_l(t)}{y_{l'}(t)}
\]

If for some \( l \) and any \( l' \neq l \), \( r_{l/l'}(t) \to \infty \), then \( y_l(t) \) dominates and \( \hat{y}_l(t) \to 1 \), i.e., the dynamics converges to \( u_l \).

Now our task is to know which initial condition of \( y_l \) and \( b_l \) makes \( r_{l/l'}(t) \to +\infty \). By comparing the critical time we know which component \( l \) shoots up the earliest and that \( l^* = \arg \min_l t_l^* \) is the winner, without computing the normalization constant \( Z(t) \).

The critical time satisfies

\[
y_l(0) y_{l'}(0) + b_l e^{b_l t_l^*} = 1
\]

so

\[
t_l^* = \frac{1}{b_l} \ln \left( 1 + \frac{b_l}{y_l(0)} \right)
\]

It is clear that when \( y_l(0) \) is larger, the critical time \( t_l^* \) becomes smaller and the \( l \)-th component becomes more advantageous over other components.  

For \( b_l > 0 \), we have:

\[
\frac{\partial t_l^*}{\partial b_l} = \frac{1}{b_l^2} \left[ \frac{b_l/y_l(0)}{1 + b_l/y_l(0)} - \ln(1 + b_l/y_l(0)) \right] < 0
\]

where the last inequality is due to the fact that \( \frac{x}{1+x} < \ln(1+x) \) for \( x > 0 \). Therefore, larger \( b_l \) leads to smaller \( t_l^* \). Since adding \( cu_l u_l \) with \( c > 0 \) to \( A_k \) increase \( b_l \), it leads to smaller \( t_l^* \) and thus increases the advantage of the \( l \)-th component.

Therefore, larger \( b_l \) and larger \( y_l(0) \) both leads to smaller \( t_l^* \). For the same \( t_l^* \), larger \( b_l \) can trade for smaller \( y_l(0) \), i.e., larger attractive basin.

**Remark. Special case.** We start by assuming only one \( \epsilon_l \neq 0 \) and all other \( \epsilon_{l'} = 0 \) for \( l' \neq l \), and then we generalize to the case when all \( \{\epsilon_l\} \) are real numbers.

To quantify the probability that a random weight initialization leads to convergence of \( u_l \), we setup some notations. Let the event \( E_l \) be “a random weight initialization of \( y \) leads to \( y \to \epsilon_l \)”, or equivalently \( w \to u_l \). Let \( Y_l \) be the random variable that instantiates the initial value of \( y_l(0) \) due to random weight initialization. Then the convergence event \( E_l \) is equivalent to the following: (1) \( Y_l > 0 \) (so that the \( i \)-component has the opportunity to grow), and (2) \( Y_l + \epsilon_l \) is the maximum over all \( Y_l \) for any \( l' \neq l \), where \( \epsilon_l \) is an advantage (\( > 0 \)) or disadvantage (\( < 0 \)) achieved by having larger/smaller \( b_l \) due to global modulation (e.g., \( c \)). Therefore, we also call \( \epsilon_l \) the modulation factor.

Here we discuss about a simple case that \( Y_l \sim U[-1, 1] \) and for \( l' \neq l \), \( Y_{l'} \) and \( Y_l \) are independent. In this case, for a given \( l \), \( \max_{l' \neq l} Y_{l'} \) is a random variable that is independent of \( Y_l \), and has cumulative density function (CDF) \( F_{\max}(x) := \mathbb{P} [\max_{l' \neq l} Y_{l'} \leq x] = F^{d-1}(x) \), where \( F(x) \) is the CDF for \( Y_l \).

Then we have:

\[
\mathbb{P}[E_l] = \mathbb{P} \left[ Y_l > 0, Y_l + \epsilon_l \geq \max_{l' \neq l} Y_{l'} \right]
\]

\[
= \int_{0}^{+\infty} \mathbb{P} \left[ \max_{l' \neq l} Y_{l'} \leq Y_l + \epsilon_l | Y_l = y_l \right] \mathbb{P}[Y_l = y_l] dy_l
\]

\[
= \int_{0}^{+\infty} F_{\max}(y_l + \epsilon_l) dF(y_l)
\]

27
When $Y_i \sim U[-1, 1]$, $F(x) = \min \{ \frac{1}{2}(x + 1), 1 \}$ has a close form and we can compute the integral:

$$P[E_i] = P\left[Y_i > 0, Y_i + \epsilon_i \geq \max_{l \neq i} Y_l\right] = \begin{cases} \frac{1}{2} \left[1 - \left(\frac{1 + \epsilon_i}{2}\right)^d\right] + \frac{\epsilon_i}{2} & 0 \leq \epsilon_i \leq 1 \\ \frac{1}{2} \left(1 + \epsilon_i\right)^d - \left(\frac{1 + \epsilon_i}{2}\right)^d & -1 < \epsilon_i < 0 \end{cases} \quad (155)$$

We can see that the modulation factor $\epsilon_l$ plays an important role in deciding the probability that $w \to u_l$:

- **No modulation.** If $\epsilon_l = 0$, then $P[E_i] \sim \frac{1}{2}$. This means that each dimension of $y$ has equal probability to be the dominant component after training;
- **Positive modulation.** If $\epsilon_l > 0$, then $P[E_i] \geq \frac{\epsilon_l}{2}$, and that particular $l$-th component has much higher probability to become the dominant component, independent of the dimensionality $d$. Furthermore, the stronger the modulation, the higher the probability becomes.
- **Negative modulation.** Finally, if $\epsilon_l < 0$, since $1 + \epsilon_l / 2 < 1$, $P[E_i] \leq \frac{1}{2}(1 + \frac{\epsilon_l}{2})^d$ decays exponentially w.r.t the dimensionality $d$.

**General case.** We then analyze cases if all $\epsilon_l$ are real numbers. Let $l^\ast = \arg \max \epsilon_l$ and $c(k)$ be the $k$-th index of $\epsilon_l$ in descending order, i.e., $c(1) = l^\ast$.

- For $l = c(1) = l^\ast$, $\epsilon_l$ is the largest over $\{\epsilon_l\}$. Since

$$P[E_{c(1)}] = P\left[Y_i \geq 0, Y_i + \epsilon_l \geq \max_{l \neq l^\ast} Y_l + \epsilon_l\right]$$

$$\geq P\left[Y_i \geq 0, Y_i + \epsilon_l - \epsilon_{c(2)} \geq \max_{l \neq l^\ast} Y_l\right]$$

where $\epsilon_{c(1)} - \epsilon_{c(2)}$ is the gap between the largest $\epsilon_l$ and second largest $\epsilon_l$. Then this case is similar to positive modulation and thus

$$P[E_{c(1)}] \geq \frac{1}{2}\left(\epsilon_{c(1)} - \epsilon_{c(2)}\right) \quad (156)$$

- For $l$ with rank $r$ (i.e., $c(r) = l$), and any $r' < r$, we have:

$$P[E_i] = P\left[Y_i \geq 0, Y_i + \epsilon_l \geq \max_{l \neq l^\ast} Y_l + \epsilon_l\right]$$

$$\leq P\left[Y_i \geq 0, Y_i + \epsilon_l \geq \max_{l \text{: } c^{-1}(l) \leq r'} Y_l + \epsilon_l\right]$$

$$\leq P\left[Y_i \geq 0, Y_i + \epsilon_l - \epsilon_{c(r')} \geq \max_{l \text{: } c^{-1}(l) \leq r'} Y_l + \epsilon_l - \epsilon_{c(r')}\right]$$

$$\leq P\left[Y_i \geq 0, Y_i + \epsilon_l - \epsilon_{c(r')} \geq \max_{l \text{: } c^{-1}(l) \leq r'} Y_l\right]$$

Then it reduces to the case of negative modulation. Therefore, we have:

$$P[E_{c(r)}] \leq \min_{r' < r} \frac{1}{r' + 1} \left(1 - \frac{\epsilon_{c(r)} - \epsilon_{c(r')}}{2}\right)^{r' + 1} \quad (157)$$

and the probability is exponentially small if $r$ is large, i.e., $\epsilon_l$ ranks low.

### C.4 Fundamental limitation of linear models

**Theorem 4** (Gradient Colinearity in linear networks). With linear activation, $W$ follows the dynamics:

$$\hat{w}_{km} = s_{km}b_k(W, V) \quad (14)$$

where $b_k(W, V) := \mathbb{C}_\alpha \left[x_k; \sum_{k', m} s_{k'm} w_{km}^\top x_{k'}\right]$ is a linear function w.r.t. $W$. As a result, (1) $\hat{w}_{km}$ are co-linear over $m$, and (2) If $s_{km} \neq 0$, from any critical point with distinct $\{w_{km}\}$, there exists a path of critical points to identical weights ($w_{km} = w_k$).
Proof. In the linear case, we have \( \tilde{x}_{km} = x_k \) since there is no gating and all \( M \) shares the same input \( x_k \). Therefore, we can write down the dynamics of \( \hat{w}_{km} \) as the following:

\[
\hat{w}_{km} = \sum_{k',m'} s_{km,k'm'} C_{\alpha} [\tilde{x}_{km}, \tilde{x}_{k'm'}] w_{k'm'} \tag{158}
\]

\[
= \sum_{k',m'} s_{km,k'm'} C_{\alpha} [x_k, x_{k'}] w_{k'm'} \tag{159}
\]

Now we use the fact that the top-level learns fast so that \( s_{km,k'm'} = s_{km} s_{k'm'} \), which gives:

\[
\hat{w}_{km} = \sum_{k'} s_{km} C_{\alpha} [x_k, \sum_{k'} s_{k'm'} w_{k'm'} x_{k'}] \tag{160}
\]

\[
= \sum_{k'} s_{km} C_{\alpha} [x_k, s_{k'm'} w_{k'm'} x_{k'}] \tag{161}
\]

Let \( b_k(W, V) := C_{\alpha} [x_k, \sum_{k'} s_{k'm'} w_{k'm'} x_{k'}] \) be a linear function of \( W \), and we have:

\[
\hat{w}_{km} = s_{km} b_k(W, V) \tag{162}
\]

Since \( b_k \) is independent of \( m \), all \( \hat{w}_{km} \) are co-linear.

For the second part, first all if \( W^* \) is a critical point, we have the following two facts:

- Since there exists \( m \) so that \( s_{km} \neq 0 \), we know that \( b_k(W^*) = 0 \);
- If \( W^* \) contains two distinct filters \( w_{k1} = \mu_1 \neq w_{k2} = \mu_2 \) covering the same receptive field \( R_k \), then by symmetry of the weights, \( W^{**} \) in which \( w_{k1} = \mu_2 \) and \( w_{k2} = \mu_1 \), is also a critical point.

Then for any \( c \in [0, 1] \), since \( b_k(W) \) is linear w.r.t. \( W \), for the linear combination \( W^c := cW^* + (1 - c)W^{**} \), we have:

\[
b_k(W^c) = b_k(cW^* + (1 - c)W^{**}) = c b_k(W^*) + (1 - c) b_k(W^{**}) = 0 \tag{163}
\]

Therefore, \( W^c \) is also a critical point, in which \( w_{k1} = c\mu_1 + (1 - c)\mu_2 \) and \( w_{k2} = (1 - c)\mu_1 + c\mu_2 \). In particular when \( c = 1/2 \), \( w_{k1} = w_{k2} \). Repeating this process for different \( m \), we could finally reach a critical point in which all \( \hat{w}_{km} = w_k \).

\[\]
**Definition 5** (mean-backprop BatchNorm). When computing backpropagated gradient through BatchNorm, we only backprop through \( \mu_k \).

This leads to a model dynamics that has a very similar form as Lemma 4.

**Lemma 6** (Dynamics with mean-backprop BatchNorm). With mean-backprop BatchNorm (Def. 5), the dynamics is:

\[
\dot{V} = VC_\alpha[f_1^{bn}], \quad \dot{w} = [(S \otimes 1_d 1_d^T) \circ C_\alpha[\tilde{x}^{bn}]] w
\]  

(166)

**Proof.** The proof is similar to Lemma 4. For \( \dot{V} \) it is the same by replacing \( f_1 \) with \( f_1^{bn} \), which is the input to the top layer.

For \( \dot{w} \), similarly we have:

\[
\dot{w}_k = \frac{\partial E}{\partial w_k} = \sum_l C_\alpha \left[ \frac{\partial f_{2l}}{\partial w_k} f_{2l} \right]
\]  

(167)

\[
= \sum_l C_\alpha \left[ v_{lk} \frac{\partial f_{1k}^{bn}}{\partial w_k} \sum_{k'} v_{lk'} f_{1k'}^{bn} \right]
\]  

(168)

\[
= \sum_l C_\alpha \left[ v_{lk} \frac{\partial f_{1k}^{bn}}{\partial w_k} \sum_{k'} v_{lk'} (f_{1k'} - \mu_{k'}) \sigma_{k'}^{-1} \right]
\]  

(169)

Note that \( C_\alpha[\cdot, \mu_{k'}, \sigma_{k'}^{-1}] = 0 \) since \( \mu_{k'} \) and \( \sigma_{k'} \) are statistics of the batch and is constant. On the other hand, for \( \frac{\partial f_{1k}^{bn}}{\partial w_k} \), we have:

\[
\frac{\partial f_{1k}^{bn}}{\partial w_k} = \frac{1}{\sigma_k} \left( \frac{\partial f_{1k}}{\partial w_k} - \frac{\partial \mu_k}{\partial w_k} \right) - \frac{f_{1k}^{bn}}{\sigma_k} \frac{\partial \sigma_k}{\partial w_k}
\]  

(170)

Note that

\[
\frac{\partial \mu_k}{\partial w_k} = E_{\text{sample}}[\tilde{x}_k]
\]  

(171)

where \( E_{\text{sample}}[\cdot] \) is the sample mean, which is a constant over the batch. Therefore \( C_\alpha[\cdot, \frac{\partial \mu_k}{\partial w_k}] = 0 \). For mean-backprop BatchNorm, since the gradient didn’t backpropagate through the variance, the second term is simply zero. Therefore, we have:

\[
\dot{w}_k = \sum_l C_\alpha \left[ v_{lk} \sigma_k^{-1} \frac{\partial f_{1k}}{\partial w_k} \sum_{k'} v_{lk'} f_{1k'} \sigma_{k'}^{-1} \right]
\]  

(172)

\[
= \sum_l C_\alpha \left[ v_{lk} \sigma_k^{-1} \tilde{x}_{k'} \sum_{k'} v_{lk'} \tilde{w}_{k'} \tilde{x}_{k'} \sigma_{k'}^{-1} \right]
\]  

(173)

\[
= \sum_{k'} s_{kk'} C_\alpha[\sigma_k^{-1} \tilde{x}_k, \sigma_{k'}^{-1} \tilde{x}_{k'}] w_{k'}
\]  

(174)

Let \( \tilde{x}_k^{bn} := \sigma_k^{-1} \tilde{x}_k \) and \( \tilde{\tilde{x}}_k^{bn} := \left[ \begin{array}{c} \tilde{x}_1^{bn} \\ \tilde{x}_2^{bn} \\ \vdots \\ \tilde{x}_K^{bn} \end{array} \right] \in \mathbb{R}^{Kd} \). The conclusion follows.\( \Box \)

**Corollary 3** (Dynamics of \( w_k \) under conditional independence and BatchNorm). Let

\[
A_k^{bn} := \nabla[\tilde{x}_k^{bn}] = \sigma_k^{-2} A_k
\]  

(175)

\[
d_k^{bn} := \sigma_k^{-2} d_k
\]  

(176)

\[
\Delta_k^{bn} := \sigma_k^{-1} \Delta_k = E[\tilde{x}_k^{bn} | z = 1] - E[\tilde{x}_k^{bn} | z = 0]
\]  

(177)

and \( \lambda^{bn} \) be the maximal eigenvalue of \( \nabla[f_1^{bn}] \). Then we have

- (1) \( \lambda^{bn} \geq \max_k d_k^{bn} \).
• (2) For $\lambda$$_{bn}$, the associated unit eigenvector is

$$s$$_{bn} := \frac{1}{Z$$_{bn}$$} \left[ \frac{w$$^T$$k \Delta$$_k$$^bn}{\lambda$$_{bn}$$ - d$$_k$$^bn} \right] \in \mathbb{R}^K,$$

where $Z$$_{bn}$$ is the normalization constant;

• (3) the dynamics of $w_k$ is given by:

$$w_k = \left[ (s$$_k$$^bn)^2 A$$_k$$^bn + \delta$$_k$$^bn (\Delta$$_k$$^bn)^T \right] w_k$$

where

$$\delta_k := \frac{p_0 p_1}{(Z$$_{bn}$$)^2 (\lambda$$_{bn}$$ - d$$_k$$^bn)^2} \sum_{k' \neq k} \frac{(w$$^T$$k' \Delta$$_{k'}$$^bn)^2}{\lambda$$_{bn}$$ - d$$_{k'}$$^bn} \geq 0$$

Proof. Similar to Theorem 5.

Remarks In the presence of BatchNorm, Lemma 5 still holds, since it only depends on the generative structure of the data. Therefore, we have

$$\sigma^2_k \rightarrow \mathbb{V}[f_k] = w$$^T$$k \mathbb{V}[x_k] w_k = d_k + p_0 p_1 \left( w$$^T$$k_\Delta_k \right)^2$$

and thus

$$d$$_{bn}^k = \sigma^2_k = d_k \rightarrow \frac{d_k}{d_k + p_0 p_1 \left( w$$^T$$k_\Delta_k \right)^2} = \frac{1}{1 + p_0 p_1 \left( w$$^T$$k_\Delta_k / \sqrt{d_k} \right)^2}$$

becomes more uniform. This is because $d_k := w$$^T$$k L_k w_k = \mathbb{E}_z [f_k | z] \geq 0$ (Eqn. 135) is approximately the variance of $f_k$, and thus $w$$^T$$k_\Delta_k / \sqrt{d_k}$ is normalized across different receptive field $k$, reducing the effect of magnitude of the input.

Since there is no much variation within $\{d_{bn}^k\}$, $\lambda$$_{bn}$$ - d$$_k$$^bn$ becomes almost constant across different receptive field $R_k$ and won’t lead to slowness of feature learning.
The function value $\zeta(w)$ changes over the unit weight $w = [\cos \theta, \sin \theta]^T$ in 2D, parameterized by $\theta \in [-\pi, \pi]$, and each vertical dotted line is a data point. Dotted line is $E_\alpha$ with linear activation $h(x) = x$, and solid line is using ReLU activation $h(x) = \max(x, 0)$. Both lines are scaled so that their maximal value is 1.

### E Additional Experiments

#### E.1 Visualization of Local Optima in 1-Layer Setting

We added a simple experiment to visualize the local maxima of $E_\alpha(w) := \frac{1}{2} C_\alpha[h(w^T x)]$, when $w = [\cos \theta, \sin \theta]^T$ is a 2D unit vector parameterized by $\theta$, and $h$ is ReLU activation. For simplicity, here we use a uniform $\alpha := 1$.

We put a few data points $\{x_i\}$ on the unit circle, which are also parameterized by $\theta$. The data points are located at $\{\frac{-4\pi}{5}, \frac{-2\pi}{5}, \frac{-\pi}{5}, 0, \frac{\pi}{5}, \frac{2\pi}{5}, \frac{3\pi}{5}, \frac{4\pi}{5}\}$ and no data augmentation is used. The objective function $E_\alpha(\theta)$ is plotted in Fig. 8.

From the figure, we can see many local maxima ($\geq 8$) caused by nonlinearity (solid line), much more than $2 \times 2 = 4$, the maximal possible number of local maxima counting all PCA components in 2D case (i.e., $\pm \phi_1$ and $\pm \phi_2$, where $\phi_1$ and $\phi_2$ are orthogonal PCA directions in this 2D example). Moreover, unlike PCA directions, these local optima are not orthogonal to each other.

On the other hand, in the linear case (dotted line), the curve is much smoother. There are only two local maxima corresponding to $\pm \phi_2$, where $\phi_2$ is the largest PCA eigenvector.

#### E.2 2-Layer Setting

We also do more experiments on the 2-layer setting, to further verify our theoretical findings.

**Overall matching score $\bar{\chi} +$ and overall irrelevant-matching score $\bar{\chi} -$.** As defined in the main text (Eqn. 16), the matching score $\chi_{+}(R_k)$ is the degree of matching between learned weights and the embeddings of the subset $R_k^+$ of tokens that are allowed in the global patterns at each receptive field $R_k$. And the overall matching score $\bar{\chi}_{+}$ is $\bar{\chi}_{+}$ averaged over all receptive fields:

$$\chi_{+}(R_k) = \frac{1}{P} \sum_{a \in R_k^+} \max_{m} \frac{w_{km}^T u_a}{\|w_{km}\| \|u_a\|}, \quad \bar{\chi}_{+} = \frac{1}{K} \sum_{k} \chi_{+}(R_k) \tag{180}$$

Similarly, we can also define irrelevance-matching score $\chi_{-}(R_k)$ which is the degree of matching between learned weights and the embeddings of the tokens that are NOT in the subset $R_k^+$ at each receptive field $R_k$. And the overall irrelevance-matching score $\bar{\chi}_{-}$ is defined similarly.

$$\chi_{-}(R_k) = \frac{1}{P} \sum_{a \not\in R_k^+} \max_{m} \frac{w_{km}^T u_a}{\|w_{km}\| \|u_a\|}, \quad \bar{\chi}_{-} = \frac{1}{K} \sum_{k} \chi_{-}(R_k) \tag{181}$$

Ideally, we want to see high overall matching score $\bar{\chi}_{+}$ and low overall irrelevant-matching score $\bar{\chi}_{-}$, which means that the important patterns in $R_k^+$ (i.e., the patterns that are allowed in the global generators) are learned, but noisy patterns that are not part of the global patterns (i.e., the generators) are not learned. Fig. 9 shows that this indeed is the case.

**Non-uniformity $\zeta$ and how BatchNorm interacts with it.** When the scale of input data varies a lot, BatchNorm starts to matter in discovering features with low magnitude (Sec. D). To model the
scale non-uniformity, we set \( \| u_a \|_2 = \zeta \) for \( \lfloor d/2 \rfloor \) tokens and \( \| u_a \|_2 = 1/\zeta \) for the remaining tokens.

Larger \( \zeta \) corresponds to higher non-uniformity across inputs.

Fig. 11 shows that BN with ReLU activations handles large non-uniformity (large \( \zeta \)) very well, compared to the case without BN. Specifically, BN yields higher \( \bar{\chi}^+ \) in the presence of high non-uniformity (e.g., \( \zeta = 10 \)) when the network is over-parameterized (\( \beta > 1 \)) and there are multiple candidates per \( R_k \) (\( P > 1 \)), a setting that is likely to hold in real-world scenarios.

Note that in the real-world scenario, features from different channels/modalities indeed will have very different scales, and some local features that turn out to be super important to global features, can have very small scale. In such cases, normalization techniques (e.g., BatchNorm) can be very useful and our formulation justifies it in a mathematically consistent way.

**Selectively Backpropagating \( \mu_k \) and \( \sigma^2_k \) in BatchNorm.** In our analysis of BatchNorm, we assume that gradient backpropagating the mean statistics \( \mu_k \), but not variance \( \sigma^2_k \) (see Def. 5). Note that this is different from regular BatchNorm, in which both \( \mu_k \) and \( \sigma^2_k \) get backpropagated gradients. Therefore, we test how this modified BN affects the matching score \( \bar{\chi}^+ \): we change whether \( \mu_k \) and \( \sigma^2_k \) gets backpropagated gradients, while the forward pass remains the same, yielding the four following variants:

\[
\begin{align*}
\hat{f}_{\text{bn}k}[i] & := \frac{f_{\text{bn}k}[i] - \mu_k}{\sigma_k} & \text{(Vanilla BatchNorm)} \\
\hat{f}_{\text{bn}k}[i] & := \frac{f_{\text{bn}k}[i] - \text{stop-gradient}(\mu_k)}{\sigma_k} & \text{(BatchNorm with backpropated } \sigma_k) \\
\hat{f}_{\text{bn}k}[i] & := \frac{f_{\text{bn}k}[i] - \mu_k}{\text{stop-gradient}(\sigma_k)} & \text{(BatchNorm with backpropated } \mu_k) \\
\hat{f}_{\text{bn}k}[i] & := \frac{f_{\text{bn}k}[i] - \text{stop-gradient}(\mu_k)}{\text{stop-gradient}(\sigma_k)} & \text{(BatchNorm without backpropating statistics)}
\end{align*}
\]

As shown in Tbl. 1, it is interesting to see that if \( \sigma^2_k \) is not backpropagated, then the matching score \( \bar{\chi}^+ \) is actually better. This justifies our BN variant.

**Quadratic versus InfoNCE loss.** Fig. 10 shows that quadratic loss (constant pairwise importance \( \alpha \)) shows worse matching score than InfoNCE. A high-level intuition is that InfoNCE dynamically adjusts the pairwise importance \( \alpha \) (i.e., the focus of different sample pairs) during training to focus on the most important sample pairs, which makes learning patterns more efficient. We leave a comprehensive study for future work.
Figure 9: Overall matching score $\bar{\chi}_+$ (Eqn. 180, the top row) and irrelevant-matching score $\bar{\chi}_-$ (Eqn. 181) the bottom row). This is an extended version of Fig. 4 (a) When $P = 1$, linear model works well regardless of the degree of over-parameterization $\beta$, while ReLU model requires large over-parameterization to perform well; (b) When each $R_k$ has multiple local patterns that are related to the global patterns ($P > 1$) related to generators, ReLU models can capture diverse patterns better than linear ones in the over-parameterization region $\beta > 1$ and stay focus on relevant local patterns that are related to the global patterns (i.e., low $\bar{\chi}_-$). Among all activations (homogeneous or non-homogeneous), ReLU shows its strength by achieving the lowest irrelevant-matching score $\bar{\chi}_-$. In contrast, linear models are much less affected by over-parameterization. Each setting is repeated 3 times and mean/standard derivations are reported.

Figure 10: Overall matching score $\bar{\chi}_+$ (top row) and overall irrelevant-matching score $\bar{\chi}_-$ (Eqn. 16) bottom row) using quadratic loss function rather than InfoNCE. The result using InfoNCE is shown in Fig. 9 with all experiments setting being the same, except for the loss function. While we see similar trends as in Sec. 5.1 quadratic loss is not as effective as InfoNCE in feature learning.
Figure 11: The effect of BatchNorm (BN) with ReLU activation in the presence of non-uniformity $\zeta$ of the input data. The non-uniformity is set to be $\zeta = 2$ (top), $\zeta = 5$ (middle) and $\zeta = 10$ (bottom). For small non-uniformity, BN doesn’t help much. For larger non-uniformity, BN yields better matching score $\chi_+$ in the over-parameterization region (large $\beta$) and multiple tokens per RF (large $P$).

| $\beta$ | $P$ | $\mu_k$ no backprop | $\mu_k$ backprop | $\sigma_k^2$ no backprop | $\sigma_k^2$ backprop |
|---------|-----|----------------------|------------------|-----------------------|---------------------|
| 1       | 1   | 0.31 ± 0.24          | 0.23 ± 0.06      | 0.30 ± 0.24           | 0.23 ± 0.06         |
|         | 3   | 0.65 ± 0.09          | −0.03 ± 0.01     | 0.64 ± 0.07           | 0.24 ± 0.11         |
|         | 5   | 0.61 ± 0.06          | −0.00 ± 0.00     | 0.62 ± 0.02           | 0.38 ± 0.04         |
|         | 10  | 0.66 ± 0.06          | 0.53 ± 0.05      | 0.70 ± 0.03           | 0.56 ± 0.04         |
| 2       | 1   | 0.36 ± 0.13          | 0.27 ± 0.06      | 0.63 ± 0.11           | 0.33 ± 0.12         |
|         | 3   | 0.78 ± 0.01          | 0.00 ± 0.01      | 0.80 ± 0.02           | 0.41 ± 0.07         |
|         | 5   | 0.78 ± 0.02          | 0.22 ± 0.21      | 0.77 ± 0.07           | 0.56 ± 0.02         |
|         | 10  | 0.83 ± 0.08          | 0.72 ± 0.06      | 0.80 ± 0.04           | 0.71 ± 0.05         |
| 5       | 1   | 0.63 ± 0.06          | 0.43 ± 0.12      | 0.67 ± 0.06           | 0.46 ± 0.06         |
|         | 3   | 0.90 ± 0.06          | 0.45 ± 0.07      | 0.90 ± 0.03           | 0.60 ± 0.08         |
|         | 5   | 0.90 ± 0.01          | 0.72 ± 0.01      | 0.88 ± 0.06           | 0.74 ± 0.02         |
|         | 10  | 0.88 ± 0.01          | 0.88 ± 0.04      | 0.92 ± 0.03           | 0.90 ± 0.03         |
| 10      | 1   | 0.63 ± 0.12          | 0.37 ± 0.15      | 0.70 ± 0.17           | 0.46 ± 0.15         |
|         | 3   | 0.95 ± 0.05          | 0.72 ± 0.06      | 0.94 ± 0.05           | 0.80 ± 0.03         |
|         | 5   | 0.98 ± 0.02          | 0.84 ± 0.05      | 0.96 ± 0.06           | 0.92 ± 0.01         |
|         | 10  | 0.90 ± 0.01          | 0.97 ± 0.02      | 0.89 ± 0.03           | 0.97 ± 0.02         |

Table 1: The effect of backpropagating different BN statistics under non-uniformity $\zeta = 10$. Backpropagating the gradient through the sample mean $\mu_k$ but not the sample variance $\sigma_k^2$ gives overall good matching score $\chi_+$, justifying our setting of mean-backprop BatchNorm (Def. 5).
E.3 MNIST EXPERIMENTS WITH 2-LAYER SETTING

We also run the same 2-layer network (as in Sec. E.2) on MNIST [Deng (2012)] dataset. In its training, the MNIST dataset consists of 50,000 images, each with the size of 28 by 28. We split the 28-by-28 images into 4 disjoint receptive fields, each with a size of 14 by 14, just like Fig. 2(b). In each region, we vectorize the receptive field into $14 \times 14 = 196$ dimensional vector. We use smaller batchsize (8), since there are only 10 true classes in MNIST, and the probability that two samples from the same class are incorrectly treated as negative pair increases with large batchsizes. We train for 50000 minibatches and start new pass (epoch) of the dataset if needed.

Fig 12 shows the initial results. We could see that indeed filters in each receptive field capture a diverse set of pattern of the input data points (e.g., part of the digits). Furthermore, with additional data augmentation (i.e., random cropping and resizing with transform transforms.RandomResizedCrop((28, 28), scale=(0.5, 1.0), ratio=(0.5, 1.5) in PyTorch), the resulting learned patterns becomes smoother with much weaker high-frequency components. This is because the augmentation implicitly removes some of the local optima (Sec. B.5), leaving more informative local optima.
F  CONTEXT AND MOTIVATIONS OF THEOREMS

Here are the motivations and intuitions behind each major theoretical results:

• First of all, Lemma 2 and Corollary 1 try to make connection between a relatively new (and somehow abstract) concept (i.e., contrastive covariance $C_\alpha(\cdot)$) and a well-known concept (i.e., regular covariance $\nabla V(\cdot)$). This will also enable us to leverage existing properties of covariance, in order to deepen our understanding of this concept, which seems to play an important role in contrastive learning (CL).

• After that, we mainly focus on studying the CL energy function $E_\alpha$, which can be represented in terms of the contrastive covariance $C_\alpha(\cdot)$. One important property of the energy function is whether there exists any local optima and what are the properties of these local optima, since these local optima are the final destinations that network weights will converge into. Previous works in landscape analysis often talk about local optima in neural networks as abstract objects in high-dimensional space, but here, we would like to make them as concrete as possible.

• For such analysis, we always start from the simplest case (e.g., one-layer). Therefore, naturally we have Lemma 3 that characterizes critical points (as a superset of local optima), a few examples in Sec. 3.2: properties of these critical points and when they become local optima in Sec. 3. Finally, Appendix B.3 further gives a preliminary study on how the data augmentation affects the distribution of the local optima.

• Then we extend our analysis to 2-layer setting. The key question is to study the additional benefits of 2-layer network compared to $K$ independent 1-layer cases. Here the assumption of disjoint receptive fields is to make sure there is an apple-to-apple comparison, otherwise additional complexity would be involved, e.g., overlapping receptive fields. As demonstrated in Theorem 4 and Theorem 5, we find the effect of global modulation in 2-layer case, which clearly tells that the interactions across different receptive fields lead to additional terms in the dynamics that favors patterns related to latent variable $z$ that leads to conditional independence across the disjointed receptive fields.

• As a side track, in 2-layer case, we also have Theorem 6 that shows linear activation does not learn distinct features, which is consistent with 1-layer case that linear activation $h(x) = x$ only gives to maximal PCA directions (Sec. 6).

G  OTHER LEMMAS

Lemma 7 (Bound of $1 - d_t$). Define

$$
\mu(w) := \frac{1 + c(w)}{2c^2(w)} \left( \frac{\lambda_2(A(w(t)))}{\lambda_1(A(w(t)))} \right)^2 = \frac{1 + c(w)}{2c^2(w)} \left[ 1 - \frac{\lambda_{\text{gap}}(A(t))}{\lambda_1(A(t))} \right]^2 \geq 0
$$

and $\mu_t := \mu(w(t))$. If $c_t > 0$ and $\lambda_1(t) > 0$, then $1 - d_t \leq \mu_t(1 - c_t)$.

Proof. We could write $d_t$:

$$
d_t = \frac{\phi_1^\top(t)\hat{w}(t+1)}{\|\hat{w}(t+1)\|^2} = \frac{\lambda_1(t)\phi_1^\top(t)\hat{w}(t)}{\sqrt{\sum_i \lambda_i^2(t) (\phi_1^\top(t)\hat{w}(t))^2}} \geq \frac{\lambda_1(t)c_t}{\sqrt{\lambda_1^2(t)c_t^2 + \lambda_2^2(t)(1 - c_t^2)}} = \frac{1}{\sqrt{1 + \left( \frac{\lambda_2(t)}{\lambda_1(t)} \right)^2 (\frac{1}{c_t^2} - 1)}}
$$

$$
= \left[ 1 + \left( \frac{\lambda_2(t)}{\lambda_1(t)} \right)^2 \left( \frac{1}{c_t^2} - 1 \right) \right]^{-1/2}
$$

$$
\geq 1 - \frac{1}{2} \left( \frac{\lambda_2(t)}{\lambda_1(t)} \right)^2 \left( \frac{1}{c_t^2} - 1 \right) = 1 - \mu_t(1 - c_t)
$$
The first inequality is due to the fact that \( \sum_{i>1} \lambda_i^2(t) (\phi_i^T(t) w(t))^2 = 1 - c_t^2 \) (Parseval’s identity). The last inequality is due to the fact that for \( x > -1 \), \( (1 + x)\alpha \geq 1 + \alpha x \) when \( \alpha \geq 1 \) or \( \alpha < 0 \) (Bernoulli’s inequality). Therefore the conclusion holds.

**Lemma 8** (Bound of weight difference). If \( c_t > 0 \) and \( \lambda_i(t) > 0 \) for all \( i \), then \( \|\delta w(t)\|_2 \leq \sqrt{2(1 + \mu_t c_t)(1 - c_t)} \)

**Proof.** First, for \( w^T(t+1) w(t) \), we have (notice that \( \lambda_i(t) \geq 0 \)):

\[
\begin{align*}
\sum_{i} \lambda_i(t) (\phi_i^T(t) w(t))^2 & = \frac{\sum_{i} \lambda_i(t) (\phi_i^T(t) w(t))^2}{\sum_{i} \lambda_i^2(t) (\phi_i^T(t) w(t))^2} \\
& \geq \frac{\lambda_1(t) c_t^2}{\sqrt{\lambda_1^2(t) c_t^2 + \lambda_2^2(t)(1 - c_t^2)}} \geq [1 - \mu_t (1 - c_t)] c_t
\end{align*}
\]

Therefore,

\[
\begin{align*}
\|w(t+1) - w(t)\|_2 &= \sqrt{2} \sqrt{1 - w^T(t) w(t+1)} \leq \sqrt{2(1 + \mu_t c_t)(1 - c_t)}
\end{align*}
\]

**Lemma 9.** Let \( \delta A = A' - A \), then the maximal eigenvector \( \phi_1 := \phi_1(A) \) and \( \phi_1' := \phi_1(A') \) has the following Taylor expansion:

\[
\phi_1' = \phi_1 + \Delta \phi_1 + O(||\delta A||_2^2)
\]

where \( \lambda_i \) is the \( i \)-th eigenvalue of \( A \), \( \Delta \phi_1 := \sum_{j>1} \frac{\phi_j^T \delta A \phi_1}{\lambda_i - \lambda_j} \phi_j \) is the first-order term of eigenvector perturbation. In terms of inequality, there exist \( \kappa > 0 \) so that:

\[
\|\phi_1' - (\phi_1 + \Delta \phi_1)\|_2 \leq \kappa \|\delta A\|_2^2
\]

**Proof.** See time-independent perturbation theory in Quantum Mechanics [Fernández, 2000].

**Lemma 10.** Let \( L \) be the minimal Lipschitz constant of \( A \) so that \( \|A(w') - A(w)\|_2 \leq L \|w - w'\|_2 \) holds. If \( c_t > 0 \) and \( \lambda_i(t) > 0 \) for all \( i \), then we have:

\[
|d_i - c_{i+1}| = |(\phi_1(t) - \phi_1(t + 1))^T w(t+1)| \leq \nu_i(1 - c_t)
\]

where

\[
\nu(w) := 2\kappa L^2(1 + \mu(w)c(w)) + 2L\lambda_{\text{gap}}(Aw(t)) \sqrt{\mu(w)(1 + \mu(w)c(w))} \geq 0
\]

and \( \nu_i := \nu(w(t)) \).

**Proof.** Using Lemma 9 and the fact that \( \|w(t+1)\|_2 = 1 \), we have:

\[
|d_i - c_{i+1}| = |(\phi_1(t) - \phi_1(t + 1))^T w(t+1)| \leq |\Delta \phi_1^T(t) w(t+1)| + \kappa L^2 \|\delta w(t)\|_2
\]

where

\[
\Delta \phi_1(t) := \sum_{j>1} \frac{\phi_j^T \delta A(t) \phi_1(t)}{\lambda_1(t) - \lambda_j(t)} \phi_j(t)
\]

and \( \delta A(t) := A(t + 1) - A(t) \). For brevity, we omit all temporal notation if the quantity is evaluated at iteration \( t \). E.g., \( \delta w \) means \( \delta w(t) \) and \( \phi_1 \) means \( \phi_1(t) \).

Now we bound \( |\Delta \phi_1^T w(t+1)| \). Using Cauchy–Schwarz inequality:

\[
|\Delta \phi_1^T w(t+1)| = \left| \sum_{j>1} \left( \frac{\phi_j^T \delta A \phi_1}{\lambda_1 - \lambda_j} \right) \phi_j^T w(t+1) \right| \leq \sqrt{\sum_{j>1} \left( \frac{\phi_j^T \delta A \phi_1}{\lambda_1 - \lambda_j} \right)^2} \sqrt{\sum_{j>1} \left( \phi_j^T w(t+1) \right)^2} \leq \frac{1}{\lambda_{\text{gap}}(A)} \sqrt{\sum_{j>1} \left( \phi_j^T \delta A \phi_1 \right)^2} \sqrt{\sum_{j>1} \left( \phi_j^T w(t+1) \right)^2}
\]
Define $\omega$ by
\[
\omega := \frac{1}{\lambda_{\text{gap}}(A)}\|\delta A \phi_1\|_2 \sqrt{1 - d_t^2}
\]
(199)
Let $\nu_t := \frac{L}{\lambda_{\text{gap}}(A)}\|\delta w(t)\|_2 \sqrt{1 - d_t^2}$
(200)

Note that using $-1 \leq d_t \leq 1$ and Lemma 7, we have:
\[
\sqrt{1 - d_t^2} = \sqrt{1 + d_t} \sqrt{1 - d_t} \leq \sqrt{2(1 - d_t)} \leq \sqrt{2\mu_1(1 - c_t)}
\]
(201)
Finally using bound of weight difference (Lemma 8), we have:
\[
|d_t - c_{t+1}| \leq 2\kappa L^2(1 + \mu_t c_t)(1 - c_t) + L\lambda_{\text{gap}}^{-1} \sqrt{2(1 + \mu_t c_t)(1 - c_t)} \sqrt{1 - d_t^2}
\]
(202)
\[
\leq \nu_t(1 - c_t)
\]
(203)
Here $\nu_t := 2\kappa L^2(1 + \mu_t c_t) + 2L\lambda_{\text{gap}}^{-1}(A(t)) \sqrt{\mu_1(1 + \mu_t c_t)}$.

**Lemma 11.** Let $c_0 := c(w(0)) = w^T(0)\phi_1(A(w(0))) > 0$. Define local region $B_\gamma$:
\[
B_\gamma := \left\{ w : \|w - w(0)\|_2 \leq \frac{\sqrt{2(1 + \gamma)(1 - c_0)}}{1 - \sqrt{\gamma}} \right\}
\]
(204)
Define $\omega(w) := \mu(w) + \nu(w)$ to be the irregularity (also defined in Def. 7). If there exists $\gamma < 1$ so that
\[
\sup_{w \in B_\gamma} \omega(w) \leq \gamma,
\]
(205)
then
- The sequence $\{c_t\}$ increases monotonously and converges to 1;
- There exists $w_*$ so that $\lim_{t \to +\infty} w(t) = w_*$;
- $w_*$ is the maximal eigenvector of $A(w_*)$ and thus a fixed point of gradient update (Eqn. 7);
- For any $t$, $\|w(t) - w(0)\|_2 \leq \frac{\sqrt{2(1 + \gamma)(1 - c_0)}}{1 - \sqrt{\gamma}}$.
- $\|w_* - w(0)\|_2 \leq \frac{\sqrt{2(1 + \gamma)(1 - c_0)}}{1 - \sqrt{\gamma}}$. That is, $w_*$ is in the vicinity of the initial weight $w(0)$.

**Proof.** We first prove by induction that the following induction arguments are true for any $t$:
- $c_{t+1} \geq c_t > 0$;
- $1 - c_t \leq \gamma^t(1 - c_0)$;
- $w(t)$ is not far away from its initial value $w(0)$:
\[
\|w(t) - w(0)\|_2 \leq \sqrt{2(1 + \gamma)(1 - c_0)} \sum_{t'=0}^{t-1} \gamma^{t'/2}
\]
(206)
which suggests that $w(t) \in B_\gamma$.

**Base case ($t = 1$).** Since $1 \geq c_0 > 0$, $\mu(w) \geq 0$, and $A(w)$ is PD, applying Lemma 8 to $\|w(1) - w(0)\|_2$, it is clear that
\[
\|w(1) - w(0)\|_2 = \|\delta w(0)\|_2 \leq \sqrt{2}\sqrt{(1 + \mu_0 c_0)(1 - c_0)} \leq \sqrt{2(1 + \gamma)(1 - c_0)}
\]
(207)
Note that the last inequality is due to $\mu_0 \leq \gamma$. Note that
\[ 1 - c_1 = 1 - d_0 + d_0 - c_1 \leq 1 - d_t + |d_0 - c_1| \leq (\mu_0 + \nu_0)(1 - c_0) \leq \gamma(1 - c_0) \] (208)
and finally we have $c_1 \geq 1 - \gamma(1 - c_0) \geq c_0 > 0$. So the base case is satisfied.

**Inductive step.** Assume for $t$, the induction argument is true and thus $w(t) \in B_\gamma$. Therefore, by the condition, we know $\mu_t + \nu_t \leq \gamma$.

By Lemma 8, we know that
\[ \|w(t + 1) - w(t)\|_2 = \|\delta w(t)\|_2 = \sqrt{2(1 + \mu_t c_t)(1 - c_t)} \leq \sqrt{2(1 + \gamma)(1 - c_0)} \gamma^{t/2} \] (209)

Therefore, we know that $w(t + 1)$ also satisfies Eqn. 206:
\[
\begin{align*}
\|w(t + 1) - w(0)\|_2 & \leq \|w(t) - w(0)\|_2 + \|\delta w(t)\|_2 \\
& \leq \sqrt{2(1 + \gamma)(1 - c_0)} \sum_{t'=0}^{t-1} \gamma^{t'/2} + \gamma^{t/2} \\
& = \sqrt{2(1 + \gamma)(1 - c_0)} \sum_{t'=0}^{t} \gamma^{t'/2}
\end{align*}
\] (210)

Also we have:
\[
1 - c_{t+1} = 1 - d_t + d_t - c_{t+1} \leq 1 - d_t + |d_t - c_{t+1}| \leq (\mu_t + \nu_t)(1 - c_t) \leq \gamma(1 - c_t) \leq \gamma^{t+1}(1 - c_0)
\] (213)

and thus we have $c_{t+1} \geq 1 - \gamma(1 - c_t) \geq c_t > 0$.

Therefore, we have
\[ 1 - c_t \leq \gamma^t (1 - c_0) \to 0 \] (214)

thus $c_t$ is monotonously increasing to 1. This means that:
\[
\lim_{t \to +\infty} c_t = \lim_{t \to +\infty} \phi_1^\top(t)w(t) \to 1
\] (217)

Therefore, we can show that $w(t)$ is also convergent, by checking how fast $\|\delta w(t)\|_2$ decays:
\[ \|\delta w(t)\|_2 \leq \sqrt{2(1 + \mu_t c_t)(1 - c_t)} \leq \sqrt{2(1 + \gamma)(1 - c_0)} \gamma^{t/2} \] (218)

By Cauchy’s convergence test, $w(t) = w(0) + \sum_{t'=0}^{t-1} \delta w(t')$ also converges. Let
\[
\lim_{t \to +\infty} w(t) = w_*
\] (219)

This means that $A(w_*)w_* = \lambda_* w_*$ and thus $P_{w_*}^\perp A(w_*)w_* = 0$, i.e., $w_*$ is a fixed point of gradient update (Eqn. 7). Finally, we have:
\[
\|w(t) - w(0)\|_2 \leq \sqrt{2(1 + \gamma)(1 - c_0)} \sum_{t'=0}^{t-1} \gamma^{t'/2} \leq \frac{\sqrt{2(1 + \gamma)(1 - c_0)}}{1 - \sqrt{\gamma}}
\] (220)

Since $\|\cdot\|_2$ is continuous, we have the conclusion. \qed