FOR MANIFOLD LEARNING, DEEP NEURAL NETWORKS CAN BE LOCALITY SENSITIVE HASH FUNCTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

It is well established that training deep neural networks gives useful representations that capture essential features of the inputs. However, these representations are poorly understood in theory and practice. An important question for supervised learning is whether these representations capture features informative for classification, while filtering out non-informative noisy ones. We study this question formally by considering a generative process where each class is associated with a high-dimensional manifold and different classes define different manifolds. Each input of a class is produced using two latent vectors: (i) a "manifold identifier" $\boldsymbol{\gamma}$ and; (ii) a "transformation parameter" $\boldsymbol{\theta}$ that shifts examples along the surface of a manifold. E.g., $\boldsymbol{\gamma}$ might represent a canonical image of a dog, and $\boldsymbol{\theta}$ might stand for variations in pose or lighting. We provide theoretical evidence that neural representations can be viewed as LSH-like functions that map each input to an embedding that is a function of solely the informative $\boldsymbol{\gamma}$ and invariant to $\boldsymbol{\theta}$, effectively recovering the manifold identifier $\boldsymbol{\gamma}$. We prove that we get one-shot learning to unseen classes as one of the desirable consequences of this behavior.

1 INTRODUCTION

Deep learning is commonly viewed as a composition of a deep representation function which maps complex inputs (e.g., images or text) to useful representations in an embedding space, and a decision layer which easily separates the representations in the embedding space (Wong et al., 2021). However, what features are captured by the representation and what information is stripped away remains a mystery. In this work, we aim to extend our theoretical understanding of the possible mechanisms by which useful representations are learned. As an explanatory tool, consider the setting of image classification. Each class of images can be viewed as a set of transformations (e.g., different rotations, backgrounds, lighting conditions) on some canonical representative object (see discussion in DiCarlo & Cox (2007); Bengio (2012)). For example, in a video clip of a dog, we can think of the first frame as the canonical pose and every subsequent frame as a different point on the induced dog-manifold. We think of all such transformations as producing points on a fixed manifold; uniquely defining a class. Furthermore, applying the *same* set of transforms on different canonical images of different animals produces a collection of manifolds, one for each class, with a *shared* geometry.

In this work, we study whether neural representations, learned with supervision, are able to *invert* this mapping of latents to the inputs on a manifold. The learner is given each object's class label during training, but both the canonical object and the set of transformations are unknown. Specifically, we have access to a sample $\mathcal{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where each input $\mathbf{x}_i \in \mathbb{R}^d$ is drawn from a mixture distribution over m manifolds $M_1, ..., M_m$ sharing similar topologies (see Figure 1), and each label $y_i \in [m]$ corresponds to the manifold of $\mathbf{x}_i$. Further, each point $\mathbf{x} \in M_i$ is characterized by two latent vectors:

- $\boldsymbol{\gamma}_i$ is the manifold identifier (i.e., representing the canonical object) and so there is a one-to-one correspondence between each $\boldsymbol{\gamma}_i$ and M_i (at times, we will also use $M(\boldsymbol{\gamma}_i)$ to refer to the manifold associated with $\boldsymbol{\gamma}_i$).
- $\boldsymbol{\theta}$ is the transformation (e.g., representing the view or distortion). So if we fix $\boldsymbol{\gamma}_i$, the manifold M_i can be generated by sampling different values of $\boldsymbol{\theta}$.

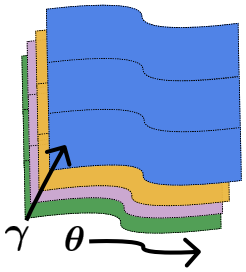

Figure 1: An illustration of our data generating process. Each class is comprised of points on a manifold; each point is characterized by two latent parameters: γ - determines the manifold the point belongs to, and $\boldsymbol{\theta}$ defines its location on the manifold.

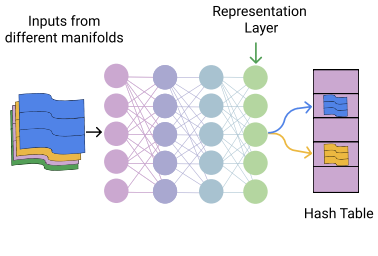

Figure 2: An illustration of a DNN as a GSH function. All points on the same manifold map to (approximately) the same representation, while any two points from different manifolds go to distant representations.

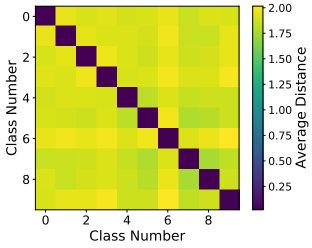

Figure 3: A confusion matrix of intra (same manifold) vs inter (different manifolds) ℓ_2-**distances** of representations for an MLP trained on synthetic data (Section 5). The intra distances are close to zero, suggesting this model is a GSH function.

A representation which, given $\mathbf{x}$ as input, filters out $\boldsymbol{\theta}$ (i.e., is invariant to change in $\boldsymbol{\theta}$) but recovers $\boldsymbol{\gamma}$ (i.e., only a function of $\boldsymbol{\gamma}$) is retaining the "useful" information [1] and is said to *invert* the manifold geometry (and the generative process). We study conditions under which DNNs are able to provably achieve this inversion.

As a warm-up, we look at how to deal with this question in the setting of (realizable) clustering. Here, $\boldsymbol{\gamma}_i$ represents the centroid of a cluster of points and $\boldsymbol{\theta}$ represents small perturbations around the centroid, so then the manifold $M(\boldsymbol{\gamma})$ is the set of all points of the form $\{\boldsymbol{\gamma} + \boldsymbol{\theta} \mid \|\boldsymbol{\theta}\| \leq \varepsilon\}$. In this setting, Locality Sensitive Hashing (LSH (Indyk & Motwani, 1998)) maps inputs so that (A) nearby inputs map to the same bucket and; (B) far away inputs go to different buckets, effectively inverting the geometry of spatial locality in this case.

More generally, when each class of input points is a complex manifold, we need something stronger. In this work, we consider a family of manifolds with a shared geometry defined using analytic functions (Section 1). We prove that DNNs with appropriate regularization, exhibit LSH-like behavior on this family of manifolds. More precisely, we show that the penultimate (i.e., representation) layer r of an appropriately trained network, will satisfy the following:

Definition 1 (Geometry Sensitive Hashing (GSH), informal)**.** *We say r is a GSH function with respect to a set of manifolds if (See Figures 2 and 3 for illustration.):*

(A) For every two points on the same manifold $\mathbf{x}_1, \mathbf{x}_2 \in M$, $\|r(\mathbf{x}_1) - r(\mathbf{x}_2)\|$ is small.
(B) For every two points on different manifolds $\mathbf{x}_1 \in M_1$, $\mathbf{x}_2 \in M_2$, $\|r(\mathbf{x}_1) - r(\mathbf{x}_2)\|$ is large.

This suggests, that DNNs whose representations satisfy the GSH property, capture the shared manifold geometry in a manner similar to how LSH functions capture spatial locality.

Motivation for Inverting the Generative Process. We can think of $\boldsymbol{\gamma}$ as a combination of semantic concepts (e.g., having a tail) thus, recovering it confers model interpretability and encourages a modular system design around representations computed in one task being passed to downstream tasks. Zimmermann et al. (2021) pursue a very similar goal in the unsupervised setting. Furthermore, a concrete consequence is representations satisfying GSH enable few-shot learning to *unseen classes* (Theorem 2) by Nearest Neighbor search on top of such representations. This of course may not be true even for models with perfect test accuracy and should motivate practitioners to study how can we achieve such representations for real datasets.

Our contributions. We summarize our contributions in a nutshell:

[1] In our model, labels provided by supervision determine what information is "useful" and what isn't. For instance, if we don't differentiate between dog breeds in our labels, then all breeds belong to the same manifold. If we use different labels for different breeds, then each breed would lie in its own sub-manifold.

- We *propose* Geometry Sensitive Hashing: a mechanism that allows us to understand how DNNs might learn good representations, and *prove* that certain DNNs properly trained on manifold data are GSH functions. Moreover, these DNNs can recover $\boldsymbol{\gamma}$ up to a linear transform thereby *inverting* the manifold geometry.
- Additionally, we argue that GSH functions have many desirable properties surrounding interpretability and aids modular design of systems which solve a complex set of tasks. As a concrete first step, we prove that GSH holds not only for train manifolds, but also for new unseen manifolds leading to effective one-shot learning.

Real-World Data. Real-world distributions are complex and cannot be described by simple analytic manifolds. Nevertheless, we investigate to what extent the GSH behavior is present in modern DNNs. We find that it continues to hold for DNNs trained on simple datasets such as MNIST. For more complex datasets such as CIFAR10, where GSH does not hold, we observe a weaker clustering effect at play. Another recent line of empirical work closely related to ours is that of (Papyan et al., 2020; Han et al., 2021) which observes a phenomenon dubbed "Neural Collapse" which encompasses our notion of GSH precisely (notion NC-1). The key difference is that they observe that the GSH property holds on train data for various deep networks such as VGG and ResNets on complex datasets such as CIFAR100 and ImageNet, however it is unclear to what extent it holds on the test distribution, which is our focus here. This indicates a possible future research direction for improving the extent to which GSH holds in deep representations at test time.

Additional Related Work. There is a rich history of works that study classification problems as manifold learning—certain notable proposals for learning manifolds include Tenenbaum et al. (2000); Belkin et al. (2006), meanwhile others such as Hein & Audibert (2005); Hein (2006) study the problem of manifold density estimation. Deep representations of complex inputs such as text and images are often used to compare the underlying objects and transfer to new classification problems (Weiss et al., 2016; Sung et al., 2018). However there is little theoretical understanding of the neural representations computed by such networks. Saunshi et al. (2019) offer theoretical insights on contrastive learning, a popular method for unsupervised learning. Khosla et al. (2020) show empirical benefits for robustness and stability when they push for deep representations to satisfy GSH. Giryes et al. (2016) attempt to study the question we do, however their proof has been shown to be false (Gulcu, 2020). Other works (Maurer et al., 2016; Du et al., 2020; Tripuraneni et al., 2020) develop a theoretical understanding of transfer learning by modeling a collection of tasks with shared parameters. Our theoretical results build on recent work exposing the benefits of wide non-linear layers (Daniely et al., 2016) and overparameterized networks (Arora et al., 2019; Allen-Zhu et al., 2018). It is also closely related to a set of works which explore the loss landscape of deep linear networks showing that all local minima are global (Ge et al., 2016; Kawaguchi, 2016). The connection between DNNs and Hash Functions was explored before (e.g., see He et al. (2021); Wang et al. (2017) and references therein). While previous works focus on empirical studies, we are able to *prove* that GSH holds for certain architectures under the manifold data assumption.

Notational Preliminaries We use $[n]$ to denote $\{1, 2, \ldots, n\}$. Boldface letters are used for vectors and capital letters denote matrices. $\mathbf{x}^\top \mathbf{y}$ or $\langle \mathbf{x}, \mathbf{y} \rangle$ denotes the inner product of $\mathbf{x}$ and $\mathbf{y}$. We use standard matrix norms:$\|A\|_F = \sqrt{\sum_{i,j} A_{ij}^2}$ is the Frobenius norm, $\|A\|_2$ is the operator or spectral norm (largest singular value of A), $\|A\|_*$ is the nuclear norm which is the sum of the singular values. $\mathbf{x} \odot \mathbf{y}$ denotes the vector obtained by point-wise multiplication of the coordinates of $\mathbf{x}$ and $\mathbf{y}$ and a similar notation is used for entry-wise multiplication of matrices as well. Given a matrix $A \in \mathbb{R}^{m \times n}$ such that $\mathrm{svd}(A) = USV^\top$, we define $A^{1/2} = US^{1/2}V^\top$. S^{d-1} is the d-dimensional unit sphere.

2 A Formal Framework for GSH

We assume that our input manifolds are sets of points in $\mathbb{R}^d$ where every manifold $M(\boldsymbol{\gamma})$ has an associated s-dimensional latent vector $\boldsymbol{\gamma} \in \mathbb{R}^s$, $s \leq d$. The manifold is then defined to be the set of points $\mathbf{x} = \mathbf{f}(\boldsymbol{\gamma}, \boldsymbol{\theta}) = (f_1(\boldsymbol{\gamma}, \boldsymbol{\theta}), \ldots, f_d(\boldsymbol{\gamma}, \boldsymbol{\theta}))$ for $\boldsymbol{\theta} \in \mathcal{T} \subseteq \mathbb{R}^k, k < d$. Here, $\mathbf{f} = \{f_i(\cdot, \cdot)\}_{i=1}^d$ is the manifold generating function where the f_i are all one-dimensional analytic functions. $\boldsymbol{\theta}$ acts as the "shift" within the manifold. Without significant loss of generality, we assume our inputs $\mathbf{x}$ and $\boldsymbol{\gamma}$s are normalized[2] and lie on S^{d-1} and S^{s-1}, the d and s-dimensional unit spheres, respectively.

[2]Note, we can make $\|\mathbf{x}\|_2 = 1$ inputs by normalization and padding by a dummy constant

When the f_is are all degree-1 polynomials we call the manifold a linear manifold. An example of a linear manifold is a $d-1$-dimensional hyperplane. Given the above generative process, we assume that there is a well-behaved analytic function to invert it. Conceptually, this function retrieves the important features and attributes of the input. (e.g., existence of nose from image of a face). Since a distinct pair of $(\boldsymbol{\gamma}, \boldsymbol{\theta})$ produce distinct images, such function always exists. For natural images and for latents we care about, this inverting function is usually non-pathological and is either analytic or easily approximable by a smooth analytic function (a small change in the pixels would not cause significant changes in the latents).

Assumption 1 (Invertibility)**.** *There is an analytic function $\mathbf{g}(\cdot) : \mathbb{R}^d \to \mathbb{R}^s$ with bounded norm Taylor expansion s.t., for every point $\mathbf{x} = \mathbf{f}(\boldsymbol{\gamma}, \boldsymbol{\theta})$ on $M(\boldsymbol{\gamma})$, $\mathbf{g}(\mathbf{x}) = \boldsymbol{\gamma}$.*

Our precise definition of the norm of an analytic function is technical and is deferred to Definition 5. We remark that it behaves similar to commonly used notions of norm in non-pathological cases. For instance, the function $g(\mathbf{x}) = e^{\boldsymbol{\beta}_1 \cdot \mathbf{x}} \cdot \sin(\boldsymbol{\beta}_2 \cdot \mathbf{x}) + \cos(\boldsymbol{\beta}_3 \cdot \mathbf{x})$ will have a constant norm if $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \boldsymbol{\beta}_3$ have constant $\|\cdot\|_2$ norm.

Train Data Generation. As described above, a set of analytic functions $\{f_i\}$ and a vector $\boldsymbol{\gamma}$ together define a manifold. We then consider a shared geometry among manifolds defined by a fixed set of $\{f_i\}$. A distribution $\mathcal{M}$ over a class of manifolds $\mathrm{supp}(\mathcal{M})$ (given by the $\{f_i\}$) is then generated by having a set Γ from which we sample $\boldsymbol{\gamma}$ associated with each manifold. We assume that all manifolds within $\mathrm{supp}(\mathcal{M})$ are reasonably separated from each other. Formally, for any two manifolds $M_1, M_2 \in \mathrm{supp}(\mathcal{M})$, we will assume that $\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2 \leq \tau$ where $\tau < 1$ is a constant[3]. Such a manifold distribution will be called τ-separated. To describe a distribution of points over a given manifold M we use the notion of a point density function $\mathcal{D}(\cdot)$ which maps a manifold M to a distribution $\mathcal{D}(M)$ over the surface of M. Training data is then generated by first drawing m manifolds $M_1, \ldots, M_m \sim \mathcal{M}$ at random. Then for each $l \in [m]$, n samples $\{(\mathbf{x}_{il}, \mathbf{y}_{il})\}_{i=1}^n$ are drawn from M_l according to the distribution $\mathcal{D}(M_l)$. Note that for convenience, we view the label $\mathbf{y}_{il}$ as a one-hot vector of length m indicating the manifold index. The learner's goal is then to learn a function which takes in these $n \times m$ pairs of $(\mathbf{x}_{il}, \mathbf{y}_{il})$ and correctly classifies which manifold a new point $\mathbf{x}$ comes from. With the above notation, we now formally define GSH.

Definition 2 (Geometry Sensitive Hashing (GSH))**.** *Given a representation function $r : \mathbb{R}^d \to \mathbb{R}^p$, and a distribution over a manifold class $\mathcal{M}$, we say that r satisfies the (ε, ρ)-hashing property on $\mathcal{M}$ with associated point density function $\mathcal{D}$ if, for some $\rho > 1, \varepsilon > 0$,*

$$V_M(r) = \mathop{\mathbb{E}}_{\mathbf{x} \sim \mathcal{D}(M)} \left[\left\| r(\mathbf{x}) - \mathop{\mathbb{E}}_{\mathbf{x} \sim \mathcal{D}(M)} [r(\mathbf{x})] \right\|_2^2 \right] \leq \varepsilon, \tag{A}$$

for all $M \in \mathrm{supp}(\mathcal{M})$. The above states that the variance of the representation across examples of a manifold is small. Moreover, for two distinct τ-separated manifolds M_1 and M_2 sampled from $\mathcal{M}$, the corresponding representations need to be far apart. That is,

$$\mathop{\mathbb{E}}_{\mathbf{x}_1 \sim \mathcal{D}(M_1), \mathbf{x}_2 \sim \mathcal{D}(M_2)} [\|r(\mathbf{x}_1) - r(\mathbf{x}_2)\|_2^2] \geq \rho\varepsilon. \tag{B}$$

A better separation factor τ between manifolds makes it easier to get a better ρ for GSH.

Our main contribution is showing that DNNs trained on manifold data can produce representations that satisfy the GSH property on the manifold distribution.

Neural Architecture. The crucial properties of a network architecture we rely on for our results are (i) over-parameterization i.e., the number of parameters is of the order of the number of train examples (or larger): this gives our neural representations the necessary expressive power and also helps optimization, (ii) at least two trainable layers in the network so that we can try to learn the representation satisfying GSH after the first trainable layer. With these points in mind, we show our theoretical results on a 3-layer network for simplifying the presentation. We consider the neural network $\hat{\mathbf{y}} = AB\sigma(C\mathbf{x})$, where the input $\mathbf{x} \in \mathbb{R}^d$ passes through a wide randomly initialized *non-trainable* layer $C \in \mathbb{R}^{D \times d}$ followed by a ReLU activation $\sigma(.)$[4]. Then, there are two trainable

[3]This is a weak assumption. In particular the average τ between two randomly sampled vectors on the unit sphere is much smaller (Claim 13).

[4]Our results hold for more general activations. The required property of an activation is that its dual should have an 'expressive' Taylor expansion. E.g., the step function or the exponential activation also satisfy this property. See Daniely et al. (2016).

fully connected layers $A \in \mathbb{R}^{m \times T}, B \in \mathbb{R}^{T \times D}$ with *no* non-linearity between them. Each row of C is drawn i.i.d. from $\mathcal{N}(\mathbf{0}, \frac{1}{D}I)$. It follows from random matrix theory that $\|C\|_2 \leq 4$ w.p. $\geq 1 - \exp(-O(D))$ (Lemma 12). This choice of architecture is guided by recent results on the expressive power of over-parameterized random ReLU layers (Daniely et al., 2016; Arora et al., 2019; Allen-Zhu et al., 2018) coupled with the fact that the loss landscape of two layer linear neural networks enjoys nice properties (Ge et al., 2016; Gunasekar et al., 2018).

Under an appropriate loss function, we prove the following Theorem (See Theorems 3 and 4 for formal versions).

Theorem 1 ((Informal) GSH holds for Manifolds from $\mathcal{M}$)**.** *For a constant τ, suppose $\mathcal{M}$ is a distribution on τ-separated manifolds with associated latent vector $\boldsymbol{\gamma} \in \mathcal{S}^{s-1}$. For any $\varepsilon > 0$, given n points samples from each of the m manifolds drawn from $\mathcal{M}$, when our 3-layer neural network of size $\mathrm{poly}(m, n, 1/\varepsilon)$ is trained on an appropriate loss, it gives a representation which satisfies the (ε, ρ)-hashing property with high probability over unseen manifolds in $\mathcal{M}$, for $\rho = \Omega(1/\varepsilon)$ when $m, n = \Theta\left(\frac{s^{O(\log(1/\varepsilon))}}{\varepsilon^2}\right)$.*

One immediate consequence of having the GSH property is transfer learning to unseen manifolds under the same shared geometry:

Theorem 2 ((Informal) GSH Implies One-Shot Learning)**.** *Given a distribution $\mathcal{M}$ over τ-separated manifolds, if a representation function $r(\cdot)$ satisfies the (ε, ρ)-GSH, for a small ε and a large enough ρ, then we have one-shot learning. That is, there is a simple hash-table lookup algorithm $\mathcal{A}$ that can classify inputs from manifold $M_{new} \sim \mathcal{M}$ with **just one sample** with high probability.*

Theorem 2 is proved as Theorem 5 in the Appendix. In addition, we show the exact recoverability of $\boldsymbol{\gamma}$ in some settings.

Remark 1. GSH implies that the representation we have computed is isomorphic to the manifold identifier $\boldsymbol{\gamma}$. We observe empirically a simple linear transform suffices to map the representation to exactly $\boldsymbol{\gamma}$. In addition, we show theoretically as well that we are able to recover $\boldsymbol{\gamma}$, albeit only for examples on the train manifolds (Section G).

Remark 2 (Extension to Deeper Networks)**.** Using recent work (Allen-Zhu et al., 2019; Du et al., 2019b) which elaborates interpolating capabilities of deeper over-parameterized networks, we can extend Theorem 1 to deeper overparameterized networks as well. One caveat is that the generalization bounds would now depend on the Rademacher complexity of the deeper network. Pursuing such an extension is significantly more challenging but that is not the main focus of this work. To avoid complicating the exposition significantly we choose to focus on a 3-layer architecture which is already non-trivial to analyze and has the salient properties required to exhibit the desired behavior. In addition, empirical work of (Papyan et al., 2020) and others shows that deep networks can indeed exhibit GSH at least on train data for complex datasets.

Additional Notation. We use $\mathbf{z}$ to denote $\sigma(C\mathbf{x})$. For succinctness, we define X_l to be the matrix whose columns are $\{\mathbf{x}_{il}\}_{i=1}^n$, Z_l is the matrix whose columns are $\{\mathbf{z}_{il}\}_{i=1}^n$. Given the label vectors $\mathbf{y}_{il}$ and the model predictions $\hat{\mathbf{y}}_{il}$ we define Y_l and $\hat{Y}_l$ similarly. Let X be the $d \times n \times m$ rank-3 tensor which is obtained by stacking the matrices X_l for $l \in [m]$. Tensors $Y, \hat{Y}, Z$ are defined similarly. We often compute a mix of empirical averages over two distributions (i) the m train manifolds (ii) the n data points from each of the m train manifold. Given a function $t(\mathbf{x})$, let $\mathbb{E}_n[t(\mathbf{x}_l)|\boldsymbol{\gamma}_l] = \frac{1}{n}\sum_{i=1}^n t(\mathbf{x}_{il})$ and given a function $t(\boldsymbol{\gamma})$ operating on a manifold, let $\mathbb{E}_m[t(\boldsymbol{\gamma})] = \frac{1}{m}\sum_{l=1}^m t(\boldsymbol{\gamma}_l)$.

Loss function. Given the one-hot label vectors $\mathbf{y}$ and the predictions $\hat{\mathbf{y}}$ made by our model we aim to minimize a weighted square loss averaged across the m train manifolds.

$$\mathcal{L}_{A,B}(Y, \hat{Y}) = \frac{1}{m}\sum_{l=1}^m \mathop{\mathbb{E}}_n\left[\|\mathbf{w}_l \odot (\mathbf{y}_l - \hat{\mathbf{y}}_l)\|_2^2 \;\middle|\; \boldsymbol{\gamma}_l\right] = \mathop{\mathbb{E}}_m\left[\left\|W_l \odot (Y_l - \hat{Y}_l)\right\|_F^2\right], \tag{1}$$

where $\mathbf{w}_l$ is a weighting of different coordinates of $\mathbf{y} - \hat{\mathbf{y}}$ such that $\mathbf{w}_{lj} = 1/2$ if $j = l$ and $1/2(m-1)$ otherwise. Each example serves as a positive example for the class corresponding to its manifold and is a negative example for all the other $m - 1$ classes. The weighting by $\mathbf{w}_l$ ensures that the total weight on the positive and negative examples is balanced and helps exclude degenerate solutions such as the all 0s vector from achieving a low loss value. We show in Section A of the supplementary

material that a small value of our weighted square loss implies a small $0/1$ classification error and vice versa. We add ℓ_2 regularization on the weight matrices A and B. The objective is then,

$$\mathcal{L}_{A,B}(Y, \hat{Y}) + \lambda_1 \|A\|_F^2 + \lambda_2 \|B\|_F^2, \tag{2}$$

When we deal with non-linear manifolds which are harder to analyze, we will require an additional component in our regularization which we term *variance regularization* (see Section 4.2).

Empirical Variant of Intra-Manifold Variance V_M. In the subsequent sections an empirical average of $V_M(r)$, the variance of representation r over points from M, across manifolds will be of importance. We define it here. Given any function $r(\cdot)$ of $\mathbf{x} \in \mathbb{R}^d$,

$$\hat{V}_{mn}(r) = \mathop{\mathbb{E}}_{m}\left[\mathop{\mathbb{E}}_{n}\left[\left\|r(\mathbf{x}_l) - \mathop{\mathbb{E}}_{n}[r(\mathbf{x}_l)]\right\|_2^2 \middle| \boldsymbol{\gamma}_l\right]\right] \tag{3}$$

A Note on Optimization Algorithms. Standard optimization algorithms such as gradient descent are theoretically shown to converge arbitrarily close to a local optimum point in many settings even for a non-convex objective. In particular, they can provably avoid second-order stable points (saddle points) with high probability (Ge et al., 2015; Jin et al., 2018; Lee et al., 2019) for Lipschitz and smooth objectives. In addition, gradient descent on overparameterized deep networks has been shown to provably avoid saddle points of all orders (Du et al., 2019a). Relying on this understanding, we assume our optimization has arrived at a local minimum and focus on understanding the properties of this local minimum.

Proof Overview Before going into further technical details, we give a short overview of Theorem 1's proof. A wide random ReLU layer enables us to approximately express arbitrary analytic functions $\boldsymbol{\gamma} = g_{inv}(\mathbf{x})$ as linear functions of the output of the ReLU layer (Lemma 2)—in fact we show that a wide random ReLU layer is "equivalent" to a kernel that produces an infinite sequence of monomials in $\mathbf{x}$ upto an orthonormal rotation. So by approximating Y as analytic functions of $\boldsymbol{\gamma}$ we get that $Y \approx W\sigma(C\mathbf{x})$ for some W. Next, since we have two layers A, B above the ReLU layer, it is possible to get a factorization $W = AB$ such that multiplication of $\mathbf{z} = \sigma(C\mathbf{x})$ by B drops any dependence on $\boldsymbol{\theta}$ and only depends on $\boldsymbol{\gamma}$—this ensures that for that choice of A, B the representation $r(\mathbf{x}) = B\mathbf{z}$ is independent of $\boldsymbol{\theta}$ (Lemma 1). Further, given the regularization we impose, the optimal output after the B layer depends only on $\boldsymbol{\gamma}$. Moreover, at the optimum, $\|B\|_F$ is bounded and independent of m, n (even though the number of parameters in B grows with m, n); similarly the average norm of A per output, $\|A\|_F/m$, can also be made constant. We then use Rademacher complexity arguments to show that if the number of training inputs per manifold n is larger than a quantity that depends on $\|B\|_F$, then the GSH property holds not just for the training inputs but for most points on the manifold. Another set of Rademacher complexity arguments show that if m is larger than a certain value that depends on $\|B\|_F$ the hashing property will generalize to most new manifolds (Section E).

3 Properties of the Architecture leading to GSH

Recall that $\|\mathbf{x}\|_2 = 1$ for all inputs. We also append a constant $1/\sqrt{2}$ to $\mathbf{x}$ to get $\mathbf{x}' = (\mathbf{x}/\sqrt{2}, 1/\sqrt{2})$. This enables a more complete kernel representation of our random ReLU layer which will help our analysis. Given (2), we show that there exists a ground truth network which makes both the loss and the regularizer terms small. Moreover, the representation computed by this ground truth is a GSH function. This is a key component of our proof.

Lemma 1 (Existence of a Good Ground Truth)**.** *Recall that the manifold identifiers $\boldsymbol{\gamma} \in \mathcal{S}^{s-1}$. There exist ground truth matrices A^*, B^* such that for any $0 < \varepsilon \leq 1/2$,*

1. $\mathcal{L}_{A^*,B^*}(Y, \hat{Y}) \leq \varepsilon$,
2. $\|A^*\|_F^2 \leq m$, $\|B^*\|_F^2 \leq \beta = s^{O(\log(1/\varepsilon))}$,
3. $B^*\sigma(C.)$ *satisfies* $(\varepsilon, \Omega(1/\varepsilon))$*-GSH.*
4. *Hidden layer width* $T = O\left(\log(mn)\log(1/\delta)\varepsilon^{-1}\right)$,

To show that the weighted square loss and the regularizer terms are small, we lean on insights from Section 3.1 which presents the power of having a random wide ReLU layer as our first layer. Once we have the bounds on $\mathcal{L}_{\hat{A},\hat{B}}(Y, \hat{Y})$ and $\|A\|_F$, property (B) for our representation follows. Our choice

of B^* will have a small intra-class representation variance averaged over the train manifolds giving us property (A). Finally to get a bound on the number of columns in A^*, we use the observation that given an A^* with a large number of columns we could use a random projection to project it down to a smaller matrix without perturbing A^*'s output by much.

3.1 KERNEL VIEW OF A NON-LINEAR RANDOM LAYER

We first show that having a wide random ReLU layer is approximately the same as an orthogonal transform on a vector whose coordinate are computed by the terms in a Taylor series expansion of a function of the input.

Claim 2. *For any $\varepsilon, \delta > 0$, and for $k \geq O((mn/\varepsilon)^{2/3})$ if the width $D \geq \Theta\left(\frac{\sqrt{mn}\log(mn/\delta)}{\varepsilon}\right)$, then, w.p $\geq 1-\delta$, there exists a matrix with $U \in \mathbb{R}^{D \times O(d^k)}$ with orthonormal rows, and $\Delta \in \mathbb{R}^{D \times mn}$, $\|\Delta\|_F < \varepsilon$, s.t., for all $i \in [n], l \in [m]$,*

$$\sigma(C\mathbf{x}_{il}) = U\left(\sqrt{\frac{1}{2\pi}}, \sqrt{\frac{1}{4}}\mathbf{x}_{il}^{\otimes 1}, \ldots, O\left(\frac{1}{k^{3/2}}\right)\mathbf{x}_{il}^{\otimes k}\right)^\top + \Delta_{il}.$$

where $\mathbf{x}^{\otimes j}$ is a vector obtained by flattening the j^{th} tensor power of the vector $\mathbf{x}$, Δ_{il} is the il^{th} column of Δ, and finally given two vectors $\mathbf{a}, \mathbf{b}$, $(\mathbf{a}, \mathbf{b})$ represents the concatenation of their coordinates to yield a single vector.

Claim 2 implies the following lemma which says that a linear function of the output of the random ReLU layer, can express bounded-norm polynomials which is used in the proof of Lemma 1.

Lemma 3. *(Informal version of Lemma 26) For $\varepsilon, \delta > 0$, and any norm bounded vector-valued analytic function $g : \mathbb{R}^d \to \mathbb{R}$ (for an appropriate notion of norm), w.p. $\geq 1-\delta$ we can approximate g using a random ReLU kernel $\sigma(C\mathbf{x})$ of width $D \geq \Theta\left(\frac{\sqrt{mn}\log(mn/\delta)}{\varepsilon}\right)$ and a bounded norm vector $\mathbf{a}$, so that, for each of the mn inputs $\mathbf{x}$, $|g(\mathbf{x}) - \mathbf{a}\sigma(C\mathbf{x})| \leq \varepsilon$.*

4 PROPERTIES OF LOCAL MINIMA OF THE LOSS

In this section, we show two properties of local minima of (2). First, we show that all local minima are global (see Ge et al. (2016) for further results of this flavor).

Lemma 4 (All Local Minima are Global)**.** *All local minima are global minima for the following objective, where $O(.)$ is any convex objective:*

$$\min_{A,B} O(AB) + \lambda_1\left(\|A\|_F^2\right) + \lambda_2\left(\|B\|_F^2\right),$$

The above lemma together with Lemma 1 implies that the desirable properties of our ground truth A^*, B^* also hold at the local minima of (2). This will follow by choosing λ_1, λ_2 appropriately.

Lemma 5. *At any local minima we have that the weighted square loss $\mathcal{L}_{\hat{A},\hat{B}}(Y, \hat{Y}) \leq 3\varepsilon$.*

Next we need to show that the empirical variant of the GSH property holds for the representation $\hat{B}\sigma(C.)$. Here our approaches for linear and non-linear manifolds differ. Linear manifolds enable a more direct analysis with a plain ℓ_2-regularization. However, we need to assume certain additional conditions on the input. The result for linear manifolds acts as a warm-up to our more general result for non-linear manifolds where we have minimal assumptions but use a stronger regularizer designed to push the representation to satisfy GSH. We describe these differences in Sections 4.1 and 4.2.

4.1 GSH PROPERTY ON LINEAR TRAIN MANIFOLDS

Recall that a linear manifold is described by a set of linear functions $\{f_i\}_{i=1}^d$ which transform $\boldsymbol{\gamma}, \boldsymbol{\theta}$ to $\mathbf{x}$. An equivalent way of describing points on a linear manifold is: $\mathbf{x} = P\boldsymbol{\gamma} + Q\boldsymbol{\theta}$ for some matrices P, Q. Without a significant loss of generality we can assume that $P\boldsymbol{\gamma} \perp Q\boldsymbol{\theta}$ (Lemma 35). Given this, we can regard as our input $\tilde{\mathbf{x}} = (\boldsymbol{\gamma}', \boldsymbol{\theta}')$ where $\boldsymbol{\theta}' \in \mathbb{R}^k$ and $\boldsymbol{\gamma}' \in \mathbb{R}^{d-k}$ by doing an appropriate

rotation of axes. Here, $\boldsymbol{\gamma}', \boldsymbol{\theta}'$ play the role of original $\boldsymbol{\gamma}, \boldsymbol{\theta}$ respectively. As before we will assume that $\|\tilde{\mathbf{x}}\|_2 = 1$. We append a constant to $\tilde{\mathbf{x}}$ as before, increasing the value of it to $O(\sqrt{k})$ for a technical nuance. This constant plays the role of a bias term. The objective for linear manifolds is then,

$$\min_{A,B} \mathcal{L}_{A,B}(Y, \hat{Y}) + \lambda_1 \|A\|_F^2 + \lambda_2 \|B\|_F^2. \tag{4}$$

Lemma 4 will imply that local minima of the above objective are global. The next step is Lemma 6 which uses a simple centering argument to show that the loss decreases when the variance of the output vector across examples from a given manifold decreases.

Lemma 6 (Centering)**.** *Replacing the output of our neural network $\hat{\mathbf{y}} = AB\sigma(C\mathbf{f}(\boldsymbol{\gamma}, \boldsymbol{\theta}))$ by $\hat{\mathbf{y}}' = \mathbb{E}_n[\hat{\mathbf{y}}|\boldsymbol{\gamma}]$ will reduce the (weighted) square loss:*

$$\mathcal{L}(Y, \hat{Y}') \leq \mathcal{L}(Y, \hat{Y}) - \hat{V}_{mn}(\hat{\mathbf{y}})/2(m-1)$$

Lemma 6 implies that a smaller variance at the output layer is beneficial. In Section D.1, we argue that it is in fact beneficial to have a small variance at the representation layer as well. Next we show Lemma 7 which lets us achieve a small variance at the representation layer by shifting weights in B away from nodes corresponding to monomials which depend on $\boldsymbol{\theta}$. This change ultimately benefits the weighted square loss in a way so that $\|A\|_F$ and $\|B\|_F$ are not impacted.

Lemma 7. *Given B s.t. $\hat{V}_{mn}(B\mathbf{z}) > \omega(\varepsilon)$, there is B' s.t. $\hat{V}_{mn}(B'\mathbf{z}) \leq O(\varepsilon)$ and $\|B'\|_F \leq \|B\|_F$.*

As we saw in Claim 2, the output of $\sigma(C\tilde{\mathbf{x}})$ can be thought of as an orthonormal transform applied onto a vector whose coordinates compute monomials of $\tilde{\mathbf{x}}$. Now we can define an association between weights of B and these monomials under which we argue using Lemma 6 that shifting all weights associated with monomials involving $\boldsymbol{\theta}'$ to corresponding monomials involving just $\boldsymbol{\gamma}'$ decreases the variance without increasing $\|B\|_F$, consequently improving objective (4). Together Lemmas 6-7 give us that at any local minima of (4) the representation $\mathbf{r}$ has the minimum variance possible.

Lemma 8. *Given any local minimum $\hat{A}, \hat{B}$ of* (6)*, and $r(\mathbf{x}) = \hat{B}\sigma(C\mathbf{x})$, we have $\hat{V}_{mn}(r) = O(\varepsilon)$.*

This will imply that at any local minimum, property (A) is satisfied at least on our train set. Next we need property (B). This follows as a consequence of having a small loss and a bound on $\|\hat{A}\|_F$.

Lemma 9. *For any local minima $\hat{A}, \hat{B}$, let $r(\mathbf{x}) = \hat{B}\sigma(C\mathbf{x})$. Then,*

$$\sum_{l=1}^{m} \sum_{j=1, j\neq l}^{m} \mathop{\mathbb{E}}_{n} \left[\|r(\mathbf{x}_l) - r(\mathbf{x}_j)\|_2^2\right] \geq \Omega(m^2).$$

4.2 GSH Property on Non-linear Train Manifolds

The same argument as in Section 4.1 fails for non-linear manifolds, as we no longer have a direct association from monomials of $\mathbf{x}$ to associated monomials of same degree in $\boldsymbol{\gamma}, \boldsymbol{\theta}$ as we had before. Instead, we show the result for non-linear manifolds using different regularizer. In addition to the ℓ_2-regularization, we penalize large variance between representation belonging to the same manifold[5].

Variance Regularization. The additional regularization term we add is the empirical average of the variance V_M across our train manifolds,

$$V_{\text{reg}}(B\sigma(C\cdot)) = \frac{n}{n-1} \hat{V}_{mn}(B\sigma(C\cdot)) \tag{5}$$

The re-scaling by $n/(n-1)$ makes each term an unbiased estimator for $V_M(B\sigma(C\cdot))$ (Lemma 49). The final objective we minimize is,

$$\mathcal{L}_{A,B}(Y, \hat{Y}) + \lambda_1 \|A\|_F^2 + \lambda_2 \left(\|B\|_F^2 + V_{\text{reg}}(B\sigma(C))\right) \tag{6}$$

Remarkably, even though (6) is different from what we had before we can still show that every local minimum is a global minimum (Lemma 45). Additionally, from the fact that the ground truth representation satisfies GSH, we get that, under the ground truth, the variance regularization term is small as well. Since the global minimum achieves a smaller objective than the ground truth, by choosing λ_1, λ_2 appropriately we get that at any local minima V_{reg} is small as well.

[5]Note that this regularization is reminiscent of the loss used in contrastive unsupervised learning (Hadsell et al., 2006; Dosovitskiy et al., 2014; Chen et al., 2020).

Lemma 10. *Given any local minimum* $\hat{A}, \hat{B}$ *of* (6)*,* $V_{\text{reg}}(\hat{B}\sigma(C\cdot)) \leq O(\varepsilon)$.

Our generalization analysis presenting population variants of the bounds we saw in the previous sections above is described in detail in the Appendix (Section E).

5 EXPERIMENTS

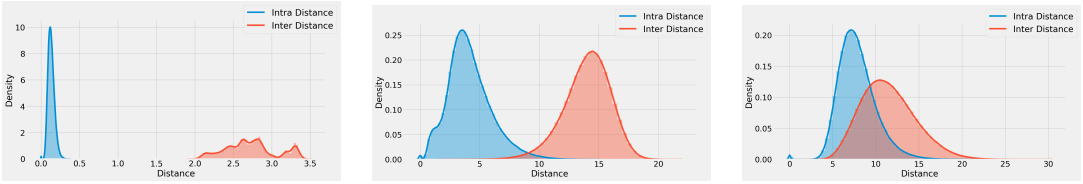

Figure 4: A comparison of intra vs inter class distances. *Left*, we train an MLP on synthetic data (see Section 5) that satisfies Assumption 1. On the *Middle* and *Right* we train a CNN on MNIST and CIFAR-10 respectively. For synthetic data, the GSH property strongly holds. The intra-distances of the representation layers for networks trained on MNIST and CIFAR-10 are also significantly smaller than their inter distances. While a case could be made that the ratio for MNIST implies GSH, for CIFAR-10 GSH does not hold. However, we do observe that a similar mechanism is partially at play.

In this section, (for full details see Appendix I), we explore to what extent the GSH property holds with today's DNNs when our assumptions are violated. In particular, we train DNNs on real and synthetic data and study the inter to intra distance ratio ρ. In contrast to the work on 'Neural Collapse' phenomenon (Papyan et al., 2020; Han et al., 2021) which shows GSH on train data, our focus is on *test samples* and *unseen classes*. We remark that the Neural Collapse line of work shows that GSH holds for train samples for deep networks (e.g., ResNet50) on complex datasets (e.g., ImageNet).

Experimental Setup We separate our experiments to two groups, based on the data source. 1. *Synthetic Data.* We randomly sample $\boldsymbol{\gamma}$ and $\boldsymbol{\theta}$ from a Multivariate Normal so that the $\boldsymbol{\gamma}$s are well separated, then chose a function satisfying Assumption 1 such as $\mathbf{f}(\cdot) = \sum_{i=1}^{4} \mathbf{f}_i(\cdot)$ where $\mathbf{f}_i$ are coordinate-wise analytic functions, e.g., (a rotation of) $\sin, \cos, \log(0.5(1+x^2))$. So a train example becomes $\mathbf{x} = \mathbf{f}(\boldsymbol{\gamma}, \boldsymbol{\theta})$ and a manifold is comprised of examples with fixed $\boldsymbol{\gamma}$ and varying $\boldsymbol{\theta}$. We train a 3-layer MLP with regularized ℓ_2 loss for 200 epochs achieving 100% train and test accuracies. 2. *Natural Images.* We train a five layer Myrtle mCNN (Page, 2018) on MNIST and CIFAR-10 using ℓ_2-regularized SGD for 50 epochs with LR of 0.1 then drop the LR to 0.01 for another 100 epochs.

Experimental Results (Figure 4). As expected, for synthetic data (see Table 1), ρ is quite large on the test data, in the range of $\rho = 10.79$-26.8 for the distributions we tried. This implies a strong GSH property and is consistent with our theory. For MNIST, $\rho = 3.36$ and for CIFAR-10 it is $\rho = 1.46$ suggesting that even though GSH doesn't hold, a weaker clustering effect exists.

One-shot Learning and $\boldsymbol{\gamma}$ Invertibility. We conduct two additional sets of experiments on synthetic data 1) We measure how well GSH holds for new manifolds (i.e., few-shot learning) and; 2) whether the learnt representation is *linearly* isomorphic to $\boldsymbol{\gamma}$. For the former, we sample an additional $50,000$ $\boldsymbol{\gamma}^{FS}$s (FS for few-shot) and generate appropriate $\mathbf{x}^{FS}$s. Then, we measure the GSH property of the representation layer. Even on these new manifolds GSH strongly holds (Figure 5) with ρ in the range of 11.09-28.77. For the latter, we use the $\{(r(\mathbf{x}_i^{FS}), \boldsymbol{\gamma}_i^{FS}\}$ as train data for a linear classifier on top of the representation produced by our MLP. We again generate a test set of never before seen $\boldsymbol{\gamma}$s. With enough samples we almost perfectly recover $\boldsymbol{\gamma}(\mathbf{x})$ from $r(\mathbf{x})$ implying a (almost) linear isomorphism between the latent and the learnt representations (see Figure 6 in the Appendix).

Conclusion and Discussion We studied how representations learnt by DNNs trained for supervised classification behave under a specific shared geometry generative process. We saw that properly trained DNNs satisfy the GSH property and recover the semantically meaningful latent representation $\boldsymbol{\gamma}$ while stripping away the dependency on the classification redundant variable $\boldsymbol{\theta}$. Notably, this is not restricted to the manifolds seen during training and thus could help shed light on how Transfer Learning works. We also observe that on real data, although today's networks do not necessarily satisfy GSH to the full extent, they exhibit weaker forms of the behavior which is valuable in itself.

REFERENCES

Naman Agarwal, Zeyuan Allen-Zhu, Brian Bullins, Elad Hazan, and Tengyu Ma. Finding approximate local minima faster than gradient descent. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1195–1199, 2017.

Atish Agarwala, Abhimanyu Das, Brendan Juba, Rina Panigrahy, Vatsal Sharan, Xin Wang, and Qiuyi Zhang. One network fits all? modular versus monolithic task formulations in neural networks. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uz5uw6gM0m.

Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. *arXiv preprint arXiv:1811.04918*, 2018.

Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pp. 242–252. PMLR, 2019.

Sanjeev Arora, Simon Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *International Conference on Machine Learning*, pp. 322–332. PMLR, 2019.

Afonso S Bandeira, Ramon Van Handel, et al. Sharp nonasymptotic bounds on the norm of random matrices with independent entries. *Annals of Probability*, 44(4):2479–2506, 2016.

Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research*, 7 (11), 2006.

Yoshua Bengio. Deep learning of representations for unsupervised and transfer learning. In *Proceedings of ICML workshop on unsupervised and transfer learning*, pp. 17–36. JMLR Workshop and Conference Proceedings, 2012.

Marcus Carlsson. Perturbation theory for the matrix square root and matrix modulus. *arXiv preprint arXiv:1810.01464*, 2018.

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.

Amit Daniely, Roy Frostig, and Yoram Singer. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. In *Advances In Neural Information Processing Systems*, pp. 2253–2261, 2016.

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

James J DiCarlo and David D Cox. Untangling invariant object recognition. *Trends in cognitive sciences*, 11(8):333–341, 2007.

Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks, 2014.

Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1675–1685. PMLR, 09–15 Jun 2019a. URL https://proceedings.mlr.press/v97/du19c.html.

Simon S. Du, Xiyu Zhai, Barnabás Póczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019b. URL https://openreview.net/forum?id=S1eK3i09YQ.

Simon S Du, Wei Hu, Sham M Kakade, Jason D Lee, and Qi Lei. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*, 2020.

Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on learning theory*, pp. 797–842. PMLR, 2015.

Rong Ge, Jason D Lee, and Tengyu Ma. Matrix completion has no spurious local minimum. *Advances in Neural Information Processing Systems*, 29:2973–2981, 2016.

Raja Giryes, Guillermo Sapiro, and Alex M. Bronstein. Deep neural networks with random gaussian weights: A universal classification strategy? *IEEE Transactions on Signal Processing*, 64(13): 3444–3457, 2016. doi: 10.1109/TSP.2016.2546221.

Talha Cihad Gulcu. Comments on "deep neural networks with random gaussian weights: A universal classification strategy?". *IEEE Transactions on Signal Processing*, 68:2401–2403, 2020.

Suriya Gunasekar, Blake Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nathan Srebro. Implicit regularization in matrix factorization. In *2018 Information Theory and Applications Workshop (ITA)*, pp. 1–10. IEEE, 2018.

Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pp. 1735–1742. IEEE, 2006.

XY Han, Vardan Papyan, and David L Donoho. Neural collapse under mse loss: Proximity to and dynamics on the central path. *arXiv preprint arXiv:2106.02073*, 2021.

Fengxiang He, Shiye Lei, Jianmin Ji, and Dacheng Tao. Neural networks behave as hash encoders: An empirical study, 2021.

Matthias Hein. Uniform convergence of adaptive graph-based regularization. In *International Conference on Computational Learning Theory*, pp. 50–64. Springer, 2006.

Matthias Hein and Jean-Yves Audibert. Intrinsic dimensionality estimation of submanifolds in rd. In *Proceedings of the 22nd international conference on Machine learning*, pp. 289–296, 2005.

Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pp. 604–613, 1998.

Chi Jin, Praneeth Netrapalli, and Michael I Jordan. Accelerated gradient descent escapes saddle points faster than gradient descent. In *Conference On Learning Theory*, pp. 1042–1085. PMLR, 2018.

Kenji Kawaguchi. Deep learning without poor local minima. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/f2fc990265c712c49d51a18a32b39f0c-Paper.pdf.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 18661–18673. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf.

Alex Krizhevsky et al. Learning multiple layers of features from tiny images. *N/A*, 2009.

Jason D Lee, Ioannis Panageas, Georgios Piliouras, Max Simchowitz, Michael I Jordan, and Benjamin Recht. First-order methods almost always avoid strict saddle points. *Mathematical programming*, 176(1):311–337, 2019.

Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. The benefit of multitask representation learning. *Journal of Machine Learning Research*, 17(81):1–32, 2016.

David Page. How to train your resnet. https://myrtle.ai/how-to-train-your-resnet-4-architecture/, 2018.

Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40): 24652–24663, 2020.

Nikunj Saunshi, Orestis Plevrakis, Sanjeev Arora, Mikhail Khodak, and Hrishikesh Khandeparkar. A theoretical analysis of contrastive unsupervised representation learning. 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. 5628–5637. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/saunshi19a.html.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1199–1208, 2018.

Michel Talagrand. Concentration of measure and isoperimetric inequalities in product spaces. *Publications Mathématiques de l'Institut des Hautes Etudes Scientifiques*, 81(1):73–205, 1995.

Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

Nilesh Tripuraneni, Michael I Jordan, and Chi Jin. On the theory of transfer learning: The importance of task diversity. *arXiv preprint arXiv:2006.11650*, 2020.

Roman Vershynin. High-dimensional probability, 2019.

Jingdong Wang, Ting Zhang, Nicu Sebe, Heng Tao Shen, et al. A survey on learning to hash. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):769–790, 2017.

Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):1–40, 2016.

Eric Wong, Shibani Santurkar, and Aleksander Mądry. Leveraging sparse linear layers for debuggable deep networks. *arXiv preprint arXiv:2105.04857*, 2021.

Roland S. Zimmermann, Yash Sharma, Steffen Schneider, Matthias Bethge, and Wieland Brendel. Contrastive learning inverts the data generating process. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12979–12990. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/zimmermann21a.html.

A ADDITIONAL PRELIMINARIES

We start the supplementary material by listing a set of additional definitions and some preliminary results. These will be for the most part statements on high-dimensional probability and linear algebra and in some cases are known from prior work or are folklore. We give the definition of analytic functions next by focusing on real-valued functions.

Definition 3 (Analytic Functions)**.** *A real-valued function $f(x)$ is an analytic function on an open set D if it is given locally by a convergent power series everywhere in D. That is, for every $x_0 \in D$,*

$$f(x) = \sum_{n=0}^{\infty} a_n (x - x_0)^n,$$

where the coefficients $a_0, a_1, \ldots$ are real numbers and the series is convergent to $f(x)$ for x in a neighborhood of x_0. We also define a norm on f as the two norm of the coefficient vector obtained when the above form is expanded to individual monomials.
Multi-variate analytic functions $f(\mathbf{x})$ are defined similarly with the difference being that the convergent power series is now a general multi-variate polynomial in the coordinates of $\mathbf{x} - \mathbf{x_0}$. The Taylor expansion can now be viewed to be of the form

$$f(\mathbf{x}) = \sum_{J} a_J \mathbf{x}^J$$

where $J = (j_1, \ldots, j_d)$ identifies the monomial $\mathbf{x}^J = x_1^{j_1} x_2^{j_2} \ldots x_d^{j_d}$.

Definition 4 (Multi-Variate Polynomials)**.** *A multi-variate polynomial $p(.)$ in $\mathbf{x} \in \mathbb{R}^d$ of degree k is defined as*

$$p(\mathbf{x}) = \sum_{J, |J| \leq k} p_J \mathbf{x}^J,$$

where $J = (J_1, \ldots, J_d)$ is a set of d integers which identifies the monomial $\mathbf{x}^J = x_1^{J_1} x_2^{J_2} \ldots x_d^{J_d}$, $|J| = \sum_{i=1}^{d} J_i$ is the degree of the monomial and p_J is the coefficient.

We will show in Section C that given an infinitely wide ReLU layer we can express any analytic function by just computing a linear function of the output of the aforementioned ReLU layer. Using this knowledge, we now present our definition of norm of an analytic function we use here.

Definition 5. *Given a multi-variate analytic function $g : \mathbb{R}^d \to \mathbb{R}$, and an infinite width ReLU layer $\sigma(C.) : \mathbb{R}^d \to \mathbb{R}^\infty$, we define*

$$\|g\| = \min_{\mathbf{a}, \mathbf{a}\sigma(C.) \equiv g} \|a\|_2.$$

This notion of the norm is chosen to help our analysis, however, our notion agrees with more common notions of norm in most non-pathological settings. We offer more intuition along with more directly interpretable bounds on our notion of the norm of an analytic function later in Section C.

Our generalization analysis uses the concepts of Rademacher complexity and its role in providing uniform convergence bounds. We give the definition of Rademacher complexity here.

Definition 6 (Rademacher Complexity)**.** *Rademacher complexity of a function class $\mathcal{F}$ is a useful quantity to understand how fast function averages for any $f \in \mathcal{F}$ converge to their mean value. Formally, the empirical Rademacher complexity of $\mathcal{F}$ on a sample set $S = (\mathbf{x_1}, \ldots, \mathbf{x_n})$ where each sample $\mathbf{x_i} \sim \mathcal{D}$, is defined as*

$$\mathcal{R}_n(\mathcal{F}) = \frac{1}{n} \mathop{\mathbb{E}}_{\boldsymbol{\xi}} \left[\sup_{f \in \mathcal{F}} \sum_{i=1}^{n} \xi_i f(\mathbf{x_i}) \right],$$

where $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_n)$ is a vector of n i.i.d. Rademacher random variables (each is $+1$ w.p. $1/2$ and -1 w.p. $1/2$). The expected Rademacher complexity is then defined as

$$\mathbb{E}[\mathcal{R}_n(\mathcal{F})] = \frac{1}{n} \mathop{\mathbb{E}}_{\boldsymbol{\xi}, \{\mathbf{x_i}\}_{i=1}^{n} \sim \mathcal{D}^n} \left[\sup_{f \in \mathcal{F}} \sum_{i=1}^{n} \xi_i f(\mathbf{x_i}) \right]$$

Given the above definition of Rademacher complexity, we have the following lemma to bound the worst deviation of the population average from the corresponding sample average over all $f \in \mathcal{F}$ (also known as uniform convergence).

Lemma 11 (Theorem 26.5 from (Shalev-Shwartz & Ben-David, 2014))**.** *Given a function class $\mathcal{F}$ of functions on inputs $\mathbf{x}$, if for all $f \in \mathcal{F}$, and for all $\mathbf{x}$, $|f(\mathbf{x})| \leq c$, we have with probability $\geq 1-\delta$,*

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}}[f(\mathbf{x})] \leq \mathop{\mathbb{E}}_{n}[f(\mathbf{x})] + 2\,\mathbb{E}[\mathcal{R}_n(\mathcal{F})] + c\sqrt{\frac{2\log(2/\delta)}{n}}.$$

The next lemma provides a bound on the spectral norm of a random matrix which is useful in bounding the norm of the weight matrix of the random ReLU layer in our analysis.

Lemma 12. *Given an $D \times d$ matrix C where each row is drawn from the d-dimensional Gaussian $\mathcal{N}(\mathbf{0}, I/D)$, we have that for a large enough constant c_1, if $D > \max(d, c_1)$, $\|C\|_2 \leq 4$ with probability $1 - \exp(-c_2 D)$ for some other constant c_2.*

Proof. Given our choice of D, the bound follows by a direct application of Corollary 3.11 from Bandeira et al. (2016) followed by simple calculations. □

We state the following two folklore results without proof.

Claim 13. *Let $\mathbf{b_1}, \mathbf{b_2}$ be two vectors sampled uniformly at random from the k-dimensional ball of unit radius S^{k-1}. Then*

$$\mathbb{P}\left[\left|\mathbf{b_1}^\top \mathbf{b_2}\right| = O(1/\sqrt{k})\right] \geq 1 - \frac{1}{\mathrm{poly}(k)}.$$

Claim 13 implies that our condition of τ-separatedness is consistent with the manifold distribution $\mathcal{M}$ being over a non-trivial number of manifolds.

Fact 14. *The Frobenius norm is invariant to multiplication by orthogonal matrices. That is, for every matrix unitary matrix U and every matrix A,*

$$\|UA\|_F = \|A\|_F$$

The following claim is also folklore.

Claim 15 (Preserving dot products with small dimensions)**.** *Let $\mathbf{x_1}, .., \mathbf{x_n}$ denote a collection of d-dimensional vectors of at most unit norm where d is large. Let $k = O(\log n \log(1/\delta)/\varepsilon^2)$ and let $R \in \mathbb{R}^{k\times d}$ be a random projection matrix whose entries are independently sampled from $\mathcal{N}(0, 1/\sqrt{k})$ that projects from d-dimensions down to k dimensions. Then R preserves all pairwise dot products $\langle \mathbf{x_i}, \mathbf{x_j}\rangle$ within additive error ε with probability $\geq 1-\delta$.*

Proof. This is a slight variant of the well-known Johnson-Lindenstrauss Lemma. We have that with probability $\geq 1-\delta$, for all $i \in [n]$, $\|R\mathbf{x_i}\|_2 = (1\pm\varepsilon)\|\mathbf{x_i}\|_2$ and $\|R(\mathbf{x}+\mathbf{y})\|_2 = (1\pm\varepsilon)\|\mathbf{x}+\mathbf{y}\|_2$. Squaring both sides gives $\langle R\mathbf{x}, R\mathbf{y}\rangle = \langle x, y\rangle \pm O(\varepsilon)$. □

We next study our notion of classification loss, namely the weighted square loss we proposed in Section 2. We relate this to the more commonly known $0/1$ classification loss now. The $0/1$ loss is defined as the fraction of mis-classified examples. We next state a lemma which shows that if our variant of the weighted square loss is really small, then the $0/1$ loss is also small. Given an m-dimensional prediction $\hat{\mathbf{y}}$ define $\mathrm{label}(\hat{\mathbf{y}}) = \mathrm{argmax}_{l\in[m]}\{\hat{\mathbf{y}}_l\}$.

Lemma 16. *Given m train manifolds, for any $\varepsilon > 0$, let ε_1 be such that $\frac{\varepsilon}{4(m-1)} > \varepsilon_1 > 0$. Then, we have over our train data,*

$$\mathcal{L}(Y, \hat{Y}) \leq \varepsilon_1 \implies \sum_{l=1}^{m}\sum_{i=1}^{n}\frac{1}{mn}\mathbb{1}(\mathrm{label}(\mathbf{y_{il}}) = \mathrm{label}(\hat{\mathbf{y}}_{\mathbf{il}})) \leq \varepsilon. \tag{7}$$

Proof. Let us focus on a single example $\mathbf{x}$ with associated true label vector $\mathbf{y}$ and predicted vector $\hat{\mathbf{y}}$. Suppose the label of $\mathbf{x}$ is l for some $l \in [m]$. Then the weighted square loss being smaller than a value $\varepsilon(\mathbf{x})$ implies

$$\left(\sum_{j\in[m],j\neq l} \frac{1}{2(m-1)}\hat{y}_j^2\right) + \frac{1}{2}(1-\hat{y}_l)^2 \leq \varepsilon(\mathbf{x}) \tag{8}$$

$$\sum_{j\in[m],j\neq l} \frac{1}{2(m-1)}\left(\hat{y}_j^2 + (1-\hat{y}_l)^2\right) \leq \varepsilon(\mathbf{x}) \tag{9}$$

$$\sum_{j\in[m],j\neq l} \frac{1}{2(m-1)}\varepsilon_{jl} \leq \varepsilon(\mathbf{x}), \tag{10}$$

where $\varepsilon_{jl} = \hat{y}_j^2 + (1-\hat{y}_l)^2$. We have $|\hat{y}_j| \leq \sqrt{\varepsilon_{jl}}$ and $|1-\hat{y}_l| \leq \sqrt{\varepsilon_{jl}} \implies \hat{y}_l \geq 1 - \sqrt{\varepsilon_{jl}}$ which implies that if $\varepsilon_{jl} < 1/4$, $\hat{y}_l > \hat{y}_j$. Now by Markov's inequality we have that the number of indices $j \neq l$ for which $\varepsilon_{jl} < 1/4$ is greater than $(m-1)(1-4\varepsilon) > (m-2)$ if $\varepsilon(\mathbf{x}) < \frac{1}{4(m-1)}$. Therefore, $\mathrm{label}(\hat{\mathbf{y}}) = l$ if $\varepsilon(\mathbf{x}) < \frac{1}{4(m-1)}$. Averaging over all examples, we have that if the average weighted square loss $\varepsilon_1 \leq \frac{\varepsilon}{4(m-1)}$, then by another application of Markov's inequality, the function $\mathrm{label}(\hat{\mathbf{y}})$ will mis-classify only an ε fraction of the train examples giving the statement of the lemma. □

Lemma 17. *Let $W \in \mathbb{R}^{m\times d}$. Then,*

$$\|W\|_1 = \min_{\substack{A,B \\ W=AB}} \frac{1}{2}\left(\|A\|_F^2 + \|B\|_F^2\right) = \min_{\substack{A,B \\ W=AB}} \|A\|_F\|B\|_F. \tag{11}$$

Proof. The proof of the Lemma is folklore and we provide it for completeness sake. Using the matrix Hölder inequality and $ab \leq \frac{1}{2}(a^2+b^2)$ we have,

$$\|W\|_1 = \|AB\|_1 \leq \|A\|_F\|B\|_F \leq \frac{1}{2}(\|A\|_F^2 + \|B\|_F^2).$$

Minimizing over A, B s.t. $W = AB$ gives one side of the inequality. On the other hand, let the singular value decomposition $W = U\Sigma V^\top$. Then, for $A = U\Sigma^{1/2}$ and $B = \Sigma^{1/2}V^T$, we have that $\|A\|_F = \|B\|_F = \|\Sigma^{1/2}\|_F = \|\Sigma^{1/2}\|_F$ so, $\|A\|_F\|B\|_F = \|\Sigma^{1/2}\|_F^2 = trace(\Sigma) = \|W\|_1$. Also, $\frac{1}{2}(\|A\|_F^2 + \|B\|_F^2) = \|\Sigma^{1/2}\|_F^2$, so $\|W\|_1 = \|A\|_F\|B\|_F = \frac{1}{2}(\|A\|_F^2 + \|B\|_F^2)$, so the inequality is tight. □

We now state a well-known claim about dual spaces.

Claim 18. *The dual of the operator norm is the trace norm and vice versa. Given two matrices A and B define $\langle A, B\rangle = Tr(A^\top B) = \sum_{i,j} A_{ij}B_{ij}$. Then,*

$$\|A\|_* = \sup_{B \text{ s.t. } \|B\|_2\leq 1} \langle A, B\rangle \tag{12}$$

$$\textit{and } \|A\|_2 = \sup_{B \text{ s.t. } \|B\|_*\leq 1} \langle A, B\rangle. \tag{13}$$

We next state a lemma which characterizes the structure of matrices A, B at any local minima of $\min_{W=AB}\|A\|_F^2 + \|B\|_F^2$. This lemma or a variant might have been used in prior work but we couldn't find a direct reference. We provide a proof here for completeness.

Lemma 19. *Let $W \in \mathbb{R}^{m\times d}$ and let $r = \min(m, d)$. Let $\hat{A}, \hat{B}$ be the minima of the following constrained optimization:*

$$\min_{W=AB} \|A\|_F^2 + \|B\|_F^2$$

then at a local minimum there is a matrix R so that if $\mathrm{svd}(W) = USV^T$ then $A = US^{1/2}R, B = R^TS^{1/2}V^T$ where $RR^T = I$. (Here if W is not full rank the SVD is written by truncating U, S, V in a way where S is square and full rank).

Proof. First assume $W = I$ and A, B are square. Then $B = A^{-1}$. So we can write $\text{svd}(A) = USV^T$ and $\text{svd}(B) = US^{-1}V^T$ where S is diagonal. Now $\|A\|_F = \|USV^T\|_F = \|S\|_F$ as multiplying by orthonormal matrix doesn't alter Frobenius norm (Fact 14). But $\|S\|_F + \|S^{-1}\|_F = \sum_i (S_{ii}^2 + 1/S_{ii}^2)$, which is minimized only when $S_{ii} = \pm 1$. So A becomes orthonormal. Further note that if that is not the case then it cannot be a local minima as there is a direction of change for some S_{ii} that improves the objective.

Next look at the case when W may not be I but is full rank and A, B are square Then let $\text{svd}(W) = USV^T$. So $AB = USV^T$. So $S^{-1/2}U^TABVS^{-1/2} = I$. Now since $S^{-1/2}U^TA$ and $BVS^{-1/2}$ are inverses of each other we can write their SVD as $U_2S_2V_2^T$ and $V_2S_2^{-1}U_2^T$. So $A = US^{1/2}U_2S_2V_2^T$ and $B = V_2S_2^{-1}U_2^TS^{1/2}V^T$. SO $|B|_F = |S_2^{-1}U_2^TS^{1/2}|_F$, and $|A|_F = |A^T|_F = |S_2U_2^TS^{1/2}|_F$. Let $Y = U_2^TS^{1/2}$. Then $|A|_F + |B|_F = |S_2Y|_F + |S_2^{-1}Y|_F = \sum_i (S_2)_{ii}^2|Y_{i,*}|_F^2 + (1/S_2)_{ii}^2|Y_{i,*}|_F^2 = \sum_i ((S_2)_{ii}^2 + (1/S_2)_{ii}^2)|Y_{i,*}|_F^2$. This is again minimized when $(S_2)_{ii} = \pm 1$ which means S_2 is orthonormal. So $A = US^{1/2}R$, and $B = R^TS^{1/2}V$ where R is orthonormal. Again note that if this is not true then some $(S_2)_{ii}$ can be perturbed and the objective can be locally improved.

The argument continues to hold if A, B are not square as the only thing that changes is that R now can become rectangular (with possibly more columns than rows) but still $RR^T = I$.

If W is not full rank again we can apply the above argument in the subspace where W has full rank (or equivalently writing SVDs in a way where the diagonal matrix S is square but U, V may be rectangular but still orthogonal). □

Corollary 20. *For any convex function $O(.)$, $\min_{A,B} O(AB) + (\|A\|_F^2 + \|B\|_F^2)$ can only be at a local minimum when $A = US^{1/2}R$ and $B = R^TS^{1/2}V^T$ for some $UU^T = I$, $RR^T = I$, $V^TV = I$*

Proof. This follows from the fact that otherwise the previous lemma can be used to alter A, B while keeping the product AB fixed and improve the regularizer part of the objective. □

B Our Theoretical Results

In this section, we state our main theorems formally. We re-state our generative process to remind the reader.

Generative Process We consider manifolds which are subsets of points in $\mathbb{R}^d$. Every manifold $M(\boldsymbol{\gamma})$ has an associated latent vector $\boldsymbol{\gamma} \in \mathbb{R}^s, s \leq d$ which acts as an identifier of $M(\boldsymbol{\gamma})$. The manifold is then defined to be the set of points $\mathbf{x} = \mathbf{f}(\boldsymbol{\gamma}, \boldsymbol{\theta}) = (f_1(\boldsymbol{\gamma}, \boldsymbol{\theta}), \ldots, f_d(\boldsymbol{\gamma}, \boldsymbol{\theta}))$ for $\boldsymbol{\theta} \in \mathcal{T} \subseteq \mathbb{R}^k, k < d$. Here, the manifold generating function $\mathbf{f} = \{f_i(\cdot, \cdot)\}_{i=1}^d$ where the f_i are all analytic functions. $\boldsymbol{\theta}$ acts as the "shift" within the manifold. Without significant loss of generality, we assume our inputs $\mathbf{x}$ and $\boldsymbol{\gamma}$s are normalized and lie on the S^{d-1} and S^{s-1} respectively. Given the above generative process, we assume that there is a well-behaved analytic function to invert it.

Assumption 2 (Invertibility: Restatement of Assumption 1)**.** *There is an analytic function $\mathbf{g}(\cdot) : \mathbb{R}^d \to \mathbb{R}^s$ with norm (Definition 5) bounded by a constant s.t. for every point $\mathbf{x} = \mathbf{f}(\boldsymbol{\gamma}, \boldsymbol{\theta})$ on $M(\boldsymbol{\gamma})$, $\mathbf{g}(\mathbf{x}) = \boldsymbol{\gamma}$.*

Next we describe how we get our train data. As described above, a set of analytic functions $\{f_i\}$ and a vector $\boldsymbol{\gamma}$ together define a manifold. A distribution $\mathcal{M}$ over a class of manifolds $\text{supp}(\mathcal{M})$ (given by the $\{f_i\}$) is then generated by having a set Γ from which we sample $\boldsymbol{\gamma}$ associated with each manifold. We assume that for any two manifolds $M_1, M_2 \in \text{supp}(\mathcal{M})$, $\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2 \leq \tau$ where $\tau < 1$ is a constant. To describe a distribution of points over a given manifold M we use the notion of a point density function $\mathcal{D}(\cdot)$ which maps a manifold M to a distribution $\mathcal{D}(M)$ over the surface of M. Training data is then generated by first drawing m manifolds $M_1, \ldots, M_m \sim \mathcal{M}$ at random. Then for each $l \in [m]$, n samples $\{(\mathbf{x}_i^l, \mathbf{y}_i^l)\}_{i=1}^n$ are drawn from M_l according to the distribution $\mathcal{D}(M_l)$. Note that we view the label $\mathbf{y}_i^l$ as a one-hot vector of length m indicating the manifold index. We consider a 3-layer neural network $\hat{\mathbf{y}} = AB\sigma(C\mathbf{x})$, where the input $\mathbf{x} \in \mathbb{R}^d$ passes through a wide randomly initialized fully-connected *non-trainable* layer $C \in \mathbb{R}^{D \times d}$ followed by a ReLU activation $\sigma(.)$. Then, there are two trainable fully connected layers $A \in \mathbb{R}^{m \times T}, B \in \mathbb{R}^{T \times D}$ with *no* non-linearity between

them. Each row of C is drawn i.i.d. from $\mathcal{N}(\mathbf{0}, \frac{1}{D}I)$. It follows from random matrix theory that $\|C\|_2 \leq 4$ w.p. $\geq 1 - \exp(-O(D))$.

Theorem 1 is our main theoretical result which is an informal variant of the following two theorems.

Theorem 3 (Main Theorem: GSH for Linear Manifolds)**.** *Let $\mathcal{M}$ be a distribution over τ-separated linear manifolds in $\mathbb{R}^d$ such that the latent vectors all lie on S^{s-1}. Given inputs $\mathbf{x}$ such that $\|\mathbf{x}\|_2 = 1$, let $\mathbf{y} = AB\sigma(C\mathbf{x})$ be the output of a 3-layer neural network, where $A \in \mathbb{R}^{m\times T}, B \in \mathbb{R}^{T\times D}$ are trainable, and $C \in \mathbb{R}^{D\times d}$ is randomly initialized as described above. Suppose we are given n data points from each of m manifolds sampled i.i.d. from $\mathcal{M}$. For any $\varepsilon > 0$, any local minima $\hat{A}, \hat{B}$ of the following loss*

$$\mathcal{L}_{A,B}(Y, \hat{Y}) + \lambda_1(\|A\|_F^2) + \lambda_2(\|B\|_F^2)$$

satisfies the following with probability $\geq 1 - \delta$,

1. *The expected weighted square loss on test samples from the train manifolds is small: $\frac{1}{m}\sum_{l=1}^{m} \mathbb{E}_{\mathbf{x}_l \sim \mathcal{D}(M_l)} \left[\|\mathbf{w_1} \odot (\mathbf{y_1} - \hat{\mathbf{y}}_\mathbf{1})\|_2^2\right] \leq O(\varepsilon)$,*
2. *The representation computed by $\hat{B}\sigma(C.)$ satisfies $(\varepsilon, 1/\varepsilon)$-GSH.*

for $n = \Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(m/\delta)}{\varepsilon^2}\right)$, $m = \Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(2/\delta)}{\varepsilon^2}\right)$ and $T = \Theta\left(\log(mn)\log(1/\delta)\varepsilon^{-1}\right), D = \Theta\left(\frac{\sqrt{mn}\log(mn/\delta)}{\varepsilon}\right)$, and $\lambda_1 = \varepsilon/m, \lambda_2 = \varepsilon/s^{O(\log(1/\varepsilon))}$.

Theorem 4 (Main Theorem: GSH for Non-Linear Manifolds)**.** *Let $\mathcal{M}$ be a distribution over τ-separated manifolds in $\mathbb{R}^d$ such that the latent vectors all lie on S^{s-1}. Given inputs $\mathbf{x}$ such that $\|\mathbf{x}\|_2 = 1$, let $\mathbf{y} = AB\sigma(C\mathbf{x})$ be the output of a 3-layer neural network, where $A \in \mathbb{R}^{m\times T}, B \in \mathbb{R}^{T\times D}$ are trainable, and $C \in \mathbb{R}^{D\times d}$ is randomly initialized as described above. Suppose we are given n data points from each of m manifolds sampled i.i.d. from $\mathcal{M}$. For any $\varepsilon > 0$, any local minima $\hat{A}, \hat{B}$ of the loss*

$$\mathcal{L}_{A,B}(Y, \hat{Y}) + \lambda_1\|A\|_F^2 + \lambda_2\left(\|B\|_F^2 + V_{reg}(B\sigma(C))\right)$$

satisfies the following with probability $\geq 1 - \delta$,

1. *The expected weighted square loss on test samples from the train manifolds is small: $\frac{1}{m}\sum_{l=1}^{m} \mathbb{E}_{\mathbf{x}_l \sim \mathcal{D}(M_l)} \left[\|\mathbf{w_1} \odot (\mathbf{y_1} - \hat{\mathbf{y}}_\mathbf{1})\|_2^2\right] \leq O(\varepsilon)$,*
2. *The representation computed by $\hat{B}\sigma(C.)$ satisfies $(\varepsilon, 1/\varepsilon)$-GSH.*

for $n = \Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(m/\delta)}{\varepsilon^2}\right)$, $m = \Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(2/\delta)}{\varepsilon^2}\right)$ and $T = \Theta\left(\log(mn)\log(1/\delta)\varepsilon^{-1}\right), D = \Theta\left(\frac{\sqrt{mn}\log(mn/\delta)}{\varepsilon}\right)$, and $\lambda_1 = \varepsilon/m, \lambda_2 = \varepsilon/s^{O(\log(1/\varepsilon))}$.

A benefit of having the hashing property is we get easy transfer learning. This was Theorem 2 in the main body. We now re-state this theorem provide its proof below.

Theorem 5 (GSH Implies One-Shot Learning)**.** *Given a distribution $\mathcal{M}$ over τ-separated manifolds, if a representation function $r(\cdot)$ satisfies the (ε, ρ)-GSH property over $\mathcal{M}$ with probability $\geq 1 - \delta$, a large enough ρ, then we have one-shot learning. That is there is a simple hash-table lookup algorithm $\mathcal{A}$ such that it learns to classify inputs from manifold $M_{new} \sim \mathcal{M}$ with just one example with probability $\geq 1 - \delta$.*

Proof. Let $\mathcal{A}$ be the following algorithm. Given a single example $\mathbf{x_{new}} \sim M_{new}$, we compute $r(\mathbf{x_{new}})$. Then given any other input $\mathbf{x}$, it does the following:

$$\begin{aligned} &\text{if} \quad \|r(\mathbf{x}) - r(\mathbf{x_{new}})\|_2^2 < 2\varepsilon, \quad \text{then} \quad \mathbf{x} \in M_{new}, \\ &\text{else} \quad \mathbf{x} \notin M_{new}. \end{aligned}$$

Since $r(\cdot)$ satisfies the (ε, ρ)-GSH w.p. $\geq 1 - \delta$, for $\rho \geq 2$, we have that $\mathcal{A}$ mis-classifies an input $\mathbf{x}$ only with probability $\leq \delta$. □

Next, it remains to prove Theorems 3 and 4. We split the proofs over multiple sections. Section C studies the properties of our architecture which is expressive enough to enable us to learn the geometry

of the manifold surfaces. Section D analyses our loss objective to show an empirical variant of the GSH for both linear and non-linear manifolds. Finally Section E is about generalizing from the empirical variant to the population variant. All three put together give us Theorems 3 and 4.

C KERNEL FUNCTION VIEW OF RANDOM LAYER WITH ACTIVATION σ

We start by looking at some properties of a wide random ReLU layer. At a high level, our goal is to show that a random ReLU layer computes a transform of the input which is highly expressive. Formally, we will show that a linear function of the feature representation computed by the random ReLU layer can approximately express 'well-behaved' analytic functions. Our formalization of what we mean by 'well-behaved' is a bit technical and relies on the understanding we develop of the transformation an input goes through via a random ReLU layer. We develop this understanding via a sequence of lemmas.

The first is the following simple lemma which focuses on a single node of a random ReLU layer and defines a kernel on the implicit feature space computed by a ReLU using the dual activation function of ReLU.

Lemma 21. *(Random ReLU Kernel) For any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ and $\mathbf{r}$ drawn from the d-dimensional normal distribution,*

$$\mathop{\mathbb{E}}_{\mathbf{r}}\left[\sigma(\mathbf{r}^\top \mathbf{x})\sigma(\mathbf{r}^\top \mathbf{y})\right] = K(\mathbf{x}, \mathbf{y}) = \|\mathbf{x}\|_2\|\mathbf{y}\|_2 \hat{\sigma}\left(\frac{\mathbf{x}^\top \mathbf{y}}{\|\mathbf{x}\|_2\|\mathbf{y}\|_2}\right).$$

where $\hat{\sigma}(\eta) = \frac{\sqrt{1-\eta^2}+(\pi-\cos^{-1}(\eta))\eta}{2\pi}$.

Proof. The result follows by noting the form of the dual activation of ReLU from Table 1 of (Daniely et al., 2016) together with the observation that for any unit vectors $\mathbf{u}, \mathbf{v}$, the joint distribution of $(\mathbf{r}^\top \mathbf{u}, \mathbf{r}^\top \mathbf{v})$ is a multivariate Gaussian with mean 0 and covariance,

$$\text{cov}(\mathbf{r}^\top \mathbf{u}, \mathbf{r}^\top \mathbf{v}) = \begin{pmatrix} \mathbf{u}^\top \\ \mathbf{v}^\top \end{pmatrix} \text{cov}(\mathbf{r})(\mathbf{u}, \mathbf{v}) = \begin{pmatrix} \|\mathbf{u}\|_2^2 & \mathbf{u}^\top \mathbf{v} \\ \mathbf{u}^\top \mathbf{v} & \|\mathbf{v}\|_2^2 \end{pmatrix} = \begin{pmatrix} 1 & \eta \\ \eta & 1 \end{pmatrix},$$

when $\mathbf{u} = \mathbf{x}/\|\mathbf{x}\|_2$ and $\mathbf{v} = \mathbf{y}/\|\mathbf{y}\|_2$. □

We let $N = mn$ denote the total number of samples we have from all our train manifolds. Recall that X denotes a rank-3 tensor of size $m \times d \times n$ obtained by stacking the X_l matrices for $l \in [m]$. In the rest of this section, we override notation and flatten X to be a $d \times N$ matrix. Given the kernel function $K(.)$ from Lemma 21, we let $K(X, X)$ be the $N \times N$ kernel matrix whose $(i, j)^{th}$ entry is $\sigma(CX_i)^\top \sigma(CX_j)$ (where X_i is the i^{th} column of X). Next, we have the following result which shows that with high probability, for any two inputs among our N train inputs, the inner product of the feature representation given at the end of a random ReLU layer is close to the kernel evaluation on this pair of inputs.

Lemma 22. *Let $N = mn$ and let $D = \Theta\left(\frac{\sqrt{N}\log(2N^2/\delta)}{\varepsilon}\right)$. Then letting $Z_D = \sigma(C \cdot X)$ where $Z_D \in \mathbb{R}^{D \times N}$, we have,*

$$\mathbb{E}\left[Z_D^\top Z_D\right] = K(X, X) \tag{14}$$

where X is the train set (and each column is of norm 1), and $K(X, X)$ is the Random ReLU kernel given in Lemma 21. Moreover, for any $\varepsilon, \delta > 0$, w.p. $\geq 1 - \delta$,

$$\|Z_D^\top Z_D - K(X, X)\|_F \leq \varepsilon$$

Proof. For the first part of the lemma, let any two indices j, k and let $\mathbf{x_i}$ and $\mathbf{x_j}$ be the appropriate columns of X. Then, $\mathbb{E}\left[Z_D^\top Z_D\right]$ in coordinate j, k is,

$$\mathbb{E}\left[Z_D^\top Z_D\right]_{j,k} = \mathbb{E}\left[\sum_{t=1}^{D} \sigma(\mathbf{r_t}^\top \mathbf{x_j})\sigma(\mathbf{r_t}^\top \mathbf{x_k})\right] = K(\mathbf{x_j}, \mathbf{x_k})$$

Where r_t is the t^{th} (random) row of C and we use linearity of expectation and Lemma 21. For the second part, write $\sigma(\mathbf{r}_t^\top \mathbf{x}_j)\sigma(\mathbf{r}_t^\top \mathbf{x}_k) \sim \frac{1}{D}\sigma(Y_t)\sigma(Z_t)$, where Y_t, Z_t are jointly distributed as $\mathcal{N}\left(0, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right)$. Now we note that $\sigma(Y_t)\sigma(Z_t)$ is a sub-exponential random variable (e.g., see (Vershynin, 2019)), either by noticing that it is a multiplication of sub-Gaussians or directly by taking any $a > 0$ and writing,

$$\mathbb{P}[\sigma(Y_t)\sigma(Z_t) > a] \leq \mathbb{P}[|Y_t| > \sqrt{a})] + \mathbb{P}[|Z_t| > \sqrt{a}] \leq 4\exp(-a/2)$$

And so, for all $a > -\mu$,

$$\mathbb{P}[\sigma(Y_t)\sigma(Z_t) - \mu > a] \leq 4\exp\left(-\frac{(a+\mu)}{2}\right)$$

Thus, using a property of the sub-exponential family, there exists some universal constant $c > 0$,

$$\mathbb{P}\left[\left|\frac{1}{D}\sum_{t=1}^{D}\mathbb{E}[\sigma(Y_t)\sigma(Z_t)] - \mu\right| > \varepsilon\right] \leq 2\exp\left(\frac{-D\varepsilon}{c}\right)$$

So by taking $D = \Theta\left(\frac{\sqrt{N}\log(2n^2/\delta)}{\varepsilon}\right)$ and using the union bound over all N^2 coordinates, we have w.p. $1-\delta$, $\|Z_D^\top Z_D - K(X,X)\|_\infty \leq \frac{\varepsilon}{\sqrt{N}}$ and thus,

$$\|Z_D^\top Z_D - K(X,X)\|_F \leq \varepsilon.$$

□

Next, we show a linear algebraic result which argues that if two sets of vectors have the same set of inner products amongst them, then they must be semi-orthogonal transforms of each other. Recall that a rectangular matrix with orthogonal columns (or rows) is called semi-orthogonal.

Lemma 23. *Let $X \in \mathbb{R}^{D\times n}$ and $Y \in \mathbb{R}^{D\times n}$ and let $X^\top X = Y^\top Y$, assuming $D \geq n$. Then there exists a semi-orthogonal matrix U with orthogonal columns such that $X = UY$.*

Proof. If Y is invertible then let $U = XY^{-1}$. Then clearly $UY = X$ and

$$U^\top U = (Y^{-1})^\top X^\top X Y^{-1} = (Y^{-1})^\top Y^\top Y Y^{-1} = I.$$

Now if Y is not invertible, then first note that $X^\top X$ and X have the same null space (as they have the same right singular vectors and the singular values for the former are squares of those of the latter), and since $X^\top X = Y^\top Y$, X and Y have the same null space. Write $U' = XY^\dagger$ where $Y^\dagger$ is the pseudoinverse of Y and let V be the identity transformation on $\ker(Y)$ (and 0 everywhere else). Then, we claim that $U = U' + V$ is an orthogonal matrix such that $X = UY$. The main point is that $U'^\top U'$ is an identity operator outside $\ker Y$ and inside $\ker Y$, V is an identity operator. To see that $X = UY$, note that for every $\mathbf{x} \in \mathbb{R}^n$, write $\mathbf{x} = \hat{\mathbf{x}} + \mathbf{x}^\perp$, decomposing $\mathbf{x}$ to span $Y \oplus \ker Y$. Then, we have,

$$UY\mathbf{x} = (XY^\dagger Y + VY)(\hat{\mathbf{x}} + \mathbf{x}^\perp) = XY^\dagger Y\hat{\mathbf{x}} = X\hat{\mathbf{x}} = X\mathbf{x},$$

where we used $X\mathbf{x}^\perp = Y\mathbf{x}^\perp = 0$, $V\mathbf{y} = 0$ for $\mathbf{y} = Y\hat{\mathbf{x}} \in \text{span}\, Y$ and $Y^\dagger Y = I$ on span Y. So, UY and X agree as transformations on all of $\mathbb{R}^n$ and therefore are the same. Now, $U^\top U = (U'+V)^\top(U'+V) = U'^\top U' + V^\top V$ as $U' \perp V$. But $U'^\top U'$ is the identity on span Y (and 0 elsewhere) and $V^\top V$ is the identity on $\ker Y$ (and 0 elsewhere) so $U^\top U = I$. □

When $X^T X$ is only approximately equal to $Y^T Y$, a weaker variant of Lemma 23 still holds.

Lemma 24. *If there a sequence of matrices $X_i, Y_i \in \mathbb{R}^{D_i\times n}$ so that $X_i^\top X_i, Y_i^\top Y_i \to A$ as $D_i \to \infty$ then $X_i = U_i Y_i + \Delta_i$ where U_i are orthonormal and $\|\Delta_i\|_F \to 0$. Precisely if $\|X_i^\top X_i - A\|_F \leq \varepsilon$ and $\|Y_i^\top Y_i - A\|_F \leq \varepsilon$, then $\|\Delta_i\|_F \leq 2\sqrt{\varepsilon}$. Although we assumed X_i, Y_i have the same number of rows, if they were different we could pad the smaller matrix with zero vectors to get them to be the same shape.*

Proof. Let $(.)^{1/2}$ denote the matrix square-root operator which is defined as follows: $A^{1/2} = U\Sigma^{1/2}V^\top$ where $svd(A) = U\Sigma V^\top$. Note that this operator is continuous. Let $B_i = (X_i^\top X_i)^{1/2}$ and $C_i = (Y_i^\top Y_i)^{1/2}$. Let us pad B_i, C_i with zero rows so that they are both of dimension $D_i \times n$ then by continuity of the square root of a matrix, if $\Delta_i' = B_i - C_i$, $\|\Delta_i'\|_F \to 0$. Note that $B_i^\top B_i = X_i^\top X_i$. Then, from Lemma 23 we have that $X_i = P_i B_i$ where the P_i are orthonormal. Similarly, we have $Y_i = Q_i C_i$ where the Q_i are orthonormal. So $X_i = P_i B_i = P_i Q_i^\top Q_i B_i = P_i Q_i^\top Q_i (C_i + \Delta_i') = P_i Q_i^\top Q_i C_i + \Delta_i = P_i Q_i^\top Y_i + \Delta_i$. Finally, $\|\Delta_i\|_F = \|P_i Q_i^\top Q_i \Delta_i'\|_F = \|\Delta_i'\|_F$ from Fact 14. Therefore, $\|\Delta_i\|_F \to 0$. Hence we have $X_i = U_i Y_i + \Delta_i$ where $U_i = P_i$ and $\|\Delta_i\|_F \to 0$.

To make this precise, we note that for two $n \times n$ square matrices U, V, $\|U^{1/2} - V^{1/2}\|_2 \leq n^{-1/2}\|U - V\|_2$ (Carlsson, 2018) and so $\|U^{1/2} - V^{1/2}\|_F \leq \|U - V\|_F$. So $\|B_i - A^{1/2}\|_F \leq \sqrt{\varepsilon}$ and $\|C_i - A^{1/2}\|_F \leq \sqrt{\varepsilon}$. and so $\|\Delta_i'\|_F \leq 2\sqrt{\varepsilon}$ and hence $\|\Delta_i\|_F \leq 2\sqrt{\varepsilon}$.

□

Now, let $\hat{\sigma}(\eta) = \frac{1}{2\pi} + \frac{1}{4}\eta + \frac{1}{4\pi}\eta^2 + \frac{1}{48\pi}\eta^3 + \ldots = q_0 + q_1\eta + q_2\eta^2 + q_3\eta^3 \ldots$ denote the Taylor series expansion of $\hat{\sigma}$, the dual activation of ReLU defined in Lemma 21. Note that q_k decays as $O\left(\frac{1}{k^{3/2}}\right)$. So for $\eta \leq 1$ we can approximate this series within ε error as long as we use at least the first $O(1/\varepsilon^{2/3})$ terms.

We will now argue, using Lemmas 23 and 24, that the output of the random ReLU layer can be viewed with good probability as approximately an orthogonal linear transformation applied on a power series $\phi(\mathbf{x})$, where $\phi(\mathbf{x}) = \left(\sqrt{q_0}, \sqrt{q_1}\mathbf{x}, \sqrt{q_2}\mathbf{x}^{\otimes 2}, \sqrt{q_3}\mathbf{x}^{\otimes 4}, \ldots\right)$, an infinite dimensional vector where $\mathbf{x}^{\otimes i}$ is a flattened tensor power i of the vector $\mathbf{x}$. Let $\phi_k(\mathbf{x}) = \left(\sqrt{q_0}, \sqrt{q_1}X_i, \sqrt{q_2}X_i^{\otimes 2}, \sqrt{q_3}X_i^{\otimes 3}, \ldots \sqrt{q_k}X_k^{\otimes k}\right)$ denote the truncation of $\phi(\mathbf{x})$ up to the k^{th} tensor powers. The following Lemma allows us to think of a random ReLU layer of high enough width as kernel layer that outputs a sequence of monomials in its inputs.

Corollary 25. *For all $\varepsilon, \delta > 0$, all $k \geq O((N/\varepsilon)^{2/3})$ if the width D of the random ReLU layer is at least $\Theta\left(\frac{\sqrt{N}\log(2(N)^2/\delta)}{\varepsilon}\right)$, then, w.p. $\geq 1 - \delta$ there exists an semi-orthonormal matrix $U \in \mathbb{R}^{D \times O(d^k)}$, and $\Delta \in \mathbb{R}^{D \times N}$, $\|\Delta\|_F < 2\sqrt{\varepsilon}$ such that, for the train matrix $X \in \mathbb{R}^{d \times N}$, for all i,*

$$\sigma(CX_i) = U\phi_k(\mathbf{x}) + \Delta_i. \tag{15}$$

where X_i is the i^{th} column of X and Δ_i the i^{th} column of Δ.

Proof. For two input vectors $\mathbf{x}, \mathbf{y}$, we have,

$$\langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle = q_0 + q_1\langle \mathbf{x}, \mathbf{y} \rangle + q_2\langle \mathbf{x}^{\otimes 2}, \mathbf{y}^{\otimes 2} \rangle + q_3\langle \mathbf{x}^{\otimes 4}, \mathbf{y}^{\otimes 4} \rangle + \ldots$$

For any $J = (J_1, \ldots, J_d) \in \mathbb{N}^d$, write a monomial $x^J = x_1^{J_1} \ldots x_d^{J_d}$ and define $|J| = \sum_k J_k$. By definition, $\mathbf{x}^{\otimes i}$ is the vector of all monomials of the form $(x^J;\ |J| = i)$ and so,

$$\langle \mathbf{x}^{\otimes i}, \mathbf{y}^{\otimes i} \rangle = \sum_{|J|=i} x^J y^J = \langle \mathbf{x}, \mathbf{y} \rangle^i,$$

where the last equality is just rearranging the terms of the power of the dot product. Therefore, we can write $\langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle = K(\mathbf{x}, \mathbf{y})$ and $\phi(X)^\top \phi(X) = K(X, X)$. Now, since $\|\mathbf{x}\|_2 \leq 1$, $\langle \phi_k(\mathbf{x}), \phi_k(\mathbf{y}) \rangle$ is a $O(1/k^{3/2})$ approximation to $\hat{\sigma}(\mathbf{x}^\top \mathbf{y})$ for all pairs $\mathbf{x}, \mathbf{y}$. Hence, we have that for $k = O((N/\varepsilon)^{2/3})$, $\|\phi_k(X)^\top \phi_k(X) - K(X, X)\|_F \leq \varepsilon$. Moreover from Lemma 22 we have that w.p. $\geq 1 - \delta$, $\|Z_D^\top Z_D - K(X, X)\|_F \leq \varepsilon$ for our chosen width D. Now we can use Lemma 24 to conclude that there exists a semi-orthogonal matrix $U \in \mathbb{R}^{D \times O(d^k)}$ and an error matrix $\Delta \in \mathbb{R}^{D \times N}$, such that,

$$\sigma(CX) = U \cdot \phi_k(X) + \Delta$$

and $\|\Delta\|_F < 2\sqrt{\varepsilon}$. □

The following Lemma quantifies the norm of a function $p(\mathbf{x})$ given as a Taylor series when expressed in terms of a random ReLU kernel. We will assume, without essential loss of generality, that in the Taylor series of the random representation $\phi(\mathbf{x})$, for every monomial x^J the corresponding coefficient

q_J is non-zero. This is because by adding a constant to our input with subsequent renormalization, i.e. $\mathbf{x}' = (\mathbf{x}/\sqrt{2}, 1/\sqrt{2})$ we can use as kernel K' where $K'(\mathbf{x}, \mathbf{y}) = K(\mathbf{x}, \mathbf{y}) + 1$ wherein all the monomials exist as the Taylor series of $\hat{\sigma}$ is non-negative (also see Agarwala et al. (2021), Corollary 3, and also Lemma 9 for a matching lower bound for expressing $p(x)$ in terms of a wide random ReLU layer for a certain distribution of inputs).

Lemma 26. *For any $\varepsilon, \delta > 0$ and multi-variate polynomial $p(\mathbf{x}) = \sum_J p_J x^J$ w.p. $\geq 1 - \delta$ we can approximate p via the application of a random ReLU kernel of large enough width followed by a dot product with a vector $\mathbf{a}$, i.e. $\mathbf{a}\sigma(C\mathbf{x})$, so that $|p(\mathbf{x}) - \mathbf{a}\sigma(C\mathbf{x})| \leq \varepsilon$ for any $\mathbf{x}$ in our train samples and $\|\mathbf{a}\|_2^2 = \sum_J p_J^2/q_J$ where q_J is the coefficient of the monomial $\mathbf{x}^J\mathbf{y}^J$ in $\hat{\sigma}(\mathbf{x}^\top\mathbf{y})$.*

Proof. This follows from Corollary 25 and taking $\mathbf{a}U$ to be the vector of the coefficients of p divided by the appropriate coefficients of the Taylor series of $\phi(\mathbf{x})$. To ensure that every monomial has a non-zero coefficient in the Taylor series of the representation $\phi(.)$, we add a bias term to our input as described in the paragraph above. □

C.1 Formalizing Bounded-Norm Analytic Functions: The q-Norm

Given the understanding developed so far, we now define a norm of an analytic functions which formalizes the intuition that we want our inverting analytic function $g(.)$ from Assumption 1 to be expressible approximately using a wide enough random ReLU layer. We use Lemma 26 to define a notion of norm for any analytic function g. Given the vector $\mathbf{q}$ of coefficients of the series $\phi(\mathbf{x})$, we will define $\|g\|_q$ to be the norm of g's approximate representation using an infinitely wide random ReLU layer. That is given an infinite dimensional vector $\mathbf{a}$ and an infinitely wide random ReLU layer, let

$$\|g\|_q = \min_{\mathbf{a}, \mathbf{a}\sigma(C.) = g} \|\mathbf{a}\|_2. \tag{16}$$

We call $\|g\|_q$ the q-norm of g. We can see that $\|g\|_q^2 \leq \sum_J g_J^2/q_J$ where g_J are coefficients of monomials in the representation of g and q_J are the coefficients of the Taylor series of $\phi(.)$. We next present Lemmas which will show that for most natural well-behaved analytic functions which to not blow up to $\pm\infty$ the q-norm is bounded (see Remark 1).

The following lemma fromAgarwala et al. (2021)bounds $\|g\|_q$ for univariate functions – there the notation $\sqrt{M_g}$ was used for $\|g\|_q$ instead just as in (Arora et al., 2019).

Theorem 6. *Agarwala et al. (2021)Let $g(y)$ be a function analytic around 0, with radius of convergence R_g. Define the* auxiliary function $\tilde{g}(y)$ *by the power series*

$$\tilde{g}(y) = \sum_{k=0}^{\infty} |a_k| y^k \tag{17}$$

where the a_k are the power series coefficients of $g(y)$. Then the function $g(\boldsymbol{\beta} \cdot \mathbf{x})$ satisfies,

$$\|g\|_q \leq \beta \tilde{g}'(\beta) + \tilde{g}(0) \tag{18}$$

if the norm $\beta \equiv \|\boldsymbol{\beta}\|_2$ is less than R_g.

The tilde function is the notion of complexity which relates to the q-norm. Informally, the tilde function makes all coefficients in the Taylor series positive. The q-norm is essentially upper bounded by the value of the derivative of function at 1 (in other words, the L1 norm of the coefficients in the Taylor series). For a multivariate function $g(\mathbf{x})$, we define its tilde function $\tilde{g}(y)$ by substituting any inner product term in $\mathbf{x}$ by a univariate y. The above theorem can then also be generalized to multivariate analytic functions:

Theorem 7. *Agarwala et al. (2021)*

Let $g(\mathbf{x})$ be a function with multivariate power series representation:

$$g(\mathbf{x}) = \sum_k \sum_{v \in V_k} a_v \prod_{i=1}^{k} (\boldsymbol{\beta}_{v,i} \cdot \mathbf{x}) \tag{19}$$

where the elements of V_k index the kth order terms of the power series. We define $\tilde{g}(y) = \sum_k \tilde{a}_k y^k$ with coefficients

$$\tilde{a}_k = \sum_{v \in V_k} |a_v| \prod_{i=1}^{k} \beta_{v,i}. \tag{20}$$

If the power series of $\tilde{g}(y)$ converges at $y = 1$ then $\|g\|_q \leq \tilde{g}'(1) + \tilde{g}(0)$.

Let $g^+(\mathbf{x})$ denote the same Taylor series as $g(\mathbf{x})$ but where all coefficients have been replaced by their absolute value. Let $\|g\|_{qu}$ denote the *upper bound* $\tilde{g}'(1) + \tilde{g}'(0)$ as in Theorem 7 which ensures that $\|g\|_q \leq \|g\|_{qu}$. The following claim is evident from the expression for $\|g\|_{qu}$.

Claim 27. *The q-norm of an analytic function g satisfies the following properties.*

- $\|g\|_q \leq \|g\|_{qu}$
- $\|g\|_{qu} = \|g^+\|_{qu}$.
- $\|g_1^+ + g_2^+\|_{qu} = \|g_1^+\|_{qu} + \|g_2^+\|_{qu}$.

Corollary 28. *If for s functions $g_1(\mathbf{x}), ., g_s(\mathbf{x})$ functions $\tilde{g_i}'(1) \leq O(1), \tilde{g_i}(1) \leq O(1)$, then $\|(\sum_i g_i(\mathbf{x}))^c\|_{qu} \leq c(O(s))^c$*

Proof. Let $f(\mathbf{x}) = (\sum_i g_i(\mathbf{x}))^c$. Then $\|f\|_q \leq \tilde{f}'(1) + \tilde{f}(0)$ where $\tilde{f}(y) = (\sum_i \tilde{g}_i(y))^c$. So $\tilde{f}'(1) = c(\sum_i \tilde{g}_i(1))^{c-1}(\sum_i \tilde{g'}_i(1)) = c(O(s))^{c-1}O(s) = c(O(s))^c$. And $\tilde{f}(0) \leq \tilde{f}(1) \leq (O(s))^c$. □

Remark 1. *Most analytic functions which do not blow up to $\pm\infty$ and are Lipschitz and smooth will have a bounded q-norm according to our definition. As a concrete example to gain intuition into q-norms of analytic functions, the function $f(\mathbf{x}) = e^{\boldsymbol{\beta}_1 \cdot \mathbf{x}} \cdot sin(\boldsymbol{\beta}_2 \cdot \mathbf{x}) + cos(\boldsymbol{\beta}_3 \cdot \mathbf{x})$ has constant q-norm if $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \boldsymbol{\beta}_2$ all have a constant norm.*

D Properties of Local Minima

In the previous section, we have seen that the representation computed by a random ReLU layer is expressive enough to approximate 'well-behaved' analytic functions. In this section we will leverage this understanding to show that (a) there are good ground truth weight matrices A^*, B^* which learn to classify our train manifolds well while satisfying the GSH property, (b) and consequently any local minima of our optimization will also be a good classifier for our train data and satisfy the GSH. We start with point (a). We will assume the that the $g()$ function satisfies the conditions of Corollary 28.

Lemma 29 (Existence of Good Ground Truth)**.** *Given data from manifolds and the 3-layer architecture as described above, there exist ground truth matrices A^*, B^* such that for any $\varepsilon_1, \varepsilon_2 > 0$, with probability $\geq 1 - \delta$,*

1. $\mathcal{L}_{A^*,B^*}(Y, \hat{Y}) \leq \varepsilon_1$,
2. $\|A^*\|_F^2 \leq m$,
3. $\|B^*\|_F^2 \leq s^{O(\log(1/\varepsilon_1))}$,
4. $\hat{V}_{mn}(B^*\sigma(C.)) \leq \varepsilon_2$.

Proof. The desired output $\mathbf{y}$ is a non-continuous function whose outputs are either 0 or 1. We will approximate each coordinate of the output $\mathbf{y}$ by a continuous polynomial. First we recall that for any two distinct $\boldsymbol{\gamma_1}$ and $\boldsymbol{\gamma_2}$ from our distribution $\mathcal{M}$ we have $\langle \boldsymbol{\gamma_1}, \boldsymbol{\gamma_2} \rangle \leq \frac{1}{\sqrt{s}}$ by assumption of τ-separatedness. For any $\varepsilon > 0$, define

$$\mu(\mathbf{u}, \mathbf{v}) = \langle \mathbf{u}, \mathbf{v} \rangle^{c \log(1/\varepsilon)}. \tag{21}$$

where $\mathbf{u}, \mathbf{v}$ are vectors representing two possible values of $\boldsymbol{\gamma}$ and $c \geq 1/\log(1/\tau)$ is a constant chosen so that $c \log(1/\varepsilon)$ is an integer. Then we have, $\mu(\mathbf{u}, \mathbf{v}) = 1$ if and only if $\mathbf{u} = \mathbf{v}$ and if $\langle \mathbf{u}, \mathbf{v} \rangle \leq \tau$, then $\mu(\mathbf{u}, \mathbf{v}) \leq \varepsilon$.

Hence, for $\mathbf{x_l}$ sampled from manifold M_l we have that $g(\mathbf{x_l}) = \boldsymbol{\gamma_l}$ and,

$$\mu(\boldsymbol{\gamma_j}, g(\mathbf{x_l})) = \begin{cases} \varepsilon \text{ (at most)} & l \neq j, \\ 1 & l = j \end{cases}$$

Let $\mathbf{y_l}^* = (\mu(\boldsymbol{\gamma_1}, g(\mathbf{x_l})), \mu(\boldsymbol{\gamma_2}, g(\mathbf{x_l})), \ldots, \mu(\boldsymbol{\gamma_m}, g(\mathbf{x_l})))$. Then we have that the weighted square loss term corresponding to $\mathbf{x_l}$ $\|\mathbf{w_l} \odot (\mathbf{y_l} - \mathbf{y_l}^*)\|_2^2 \leq \varepsilon^2/2$. Based on Corollary 25 without loss of generality, we assume that the random ReLU layer outputs the monomials $\Phi(\mathbf{x})$ in co-ordinates of $\mathbf{x}$ [6]. We will now find matrices A^*, B^* so that, $A^* B^* \sigma(C\mathbf{x_l}) = \mathbf{y_l}^*$ approximately. Then bullet *1* of the Lemma will immediately follow.

To do so, we express,

$$\mu(\mathbf{u}, \mathbf{v}) = \langle \psi(\mathbf{u}), \psi(\mathbf{v}) \rangle, \tag{22}$$

where $\psi(\mathbf{u})$ and $\psi(\mathbf{v})$ are bounded-norm vectors. We do this using the binomial expansion of $\langle \mathbf{u}, \mathbf{v} \rangle^{c \log(1/\varepsilon)}$. We can write it as a weighted sum of monomials where each monomial is a product of two similar monomials in $\mathbf{u}$ and $\mathbf{v}$. We can enumerate these monomials by their degree distribution. Let $J = (J_1, \ldots, J_d) \in \mathbb{N}^d$ denote the degree distribution of a monomial in d variables. We will use the notation $x^J = x_1^{J_1} \ldots x_d^{J_d}$ to denote such a monomial over $\mathbf{x}$. Then $|J| = \sum_k J_k$ is the degree of the monomial. The expanded expression for $\mu(\mathbf{u}, \mathbf{v})$ can be written as $\sum_{J:|J|=c\log(1/\varepsilon)} a_J u^J v^J$. This in turn can be written as a dot product of two vectors whose dimension equals the total number of monomials of degree $c \log(1/\varepsilon)$ in s variables, which is $\binom{c\log(1/\varepsilon)+s-1}{s-1} = O\left(s^{c\log(1/\varepsilon)}\right)$. So precisely, $\mu(\mathbf{u}, \mathbf{v}) = \psi(\mathbf{u}) \cdot \psi(\mathbf{v})$ where $\psi(\mathbf{u})$ is a vector whose coordinates can be indexed by the different values of J and the value at the J^{th} coordinate is $(\psi(\mathbf{u}))_J = \sqrt{a_J} u^J$. Clearly then $\langle \psi(\mathbf{u}), \psi(\mathbf{v}) \rangle = \sum_J a_J u^J v^J = \mu(\mathbf{u}, \mathbf{v})$.

We will now describe the matrices A^*, B^*. For now, assume that the random ReLU kernel $\sigma(C.)$ is of infinite width. We will choose the width of the hidden layer (number of rows in A^*) to be exactly the number of different values of J. This width can be reduced to $O(\log(mn)/\varepsilon^2)$ at the expense of an additional ε error per output coordinate of A^* as shown in Lemma 30. Given this width, we simply set the l^{th} row $A_l^* = \psi(\boldsymbol{\gamma_l})$. Then B^* is chosen such that the output of the hidden layer $r = B^* \sigma(C\mathbf{x_l}) \approx \psi(g(\mathbf{x_l}))$. To see that such a B^* exists, note that we need the J^{th} coordinate of r, $r_J = \sqrt{a_J}(g(\mathbf{x_l}))^J$. Since $g(\mathbf{x_l})$ is analytic with a bounded norm, the $\sqrt{a_J}(g(\mathbf{x_l}))^J$ are also bounded-norm analytic functions in $\mathbf{x_l}$ and so by Lemma 3 these can be expressed using a linear transform of $\sigma(C\mathbf{x_l})$ (as the width goes to infinity). So B_J^* is chosen such that $B_J^* \cdot \sigma(C\mathbf{x}) = \sqrt{a_J}(g(\mathbf{x}))^J$. Now let us look at the Frobenius norms of A^*, B^* constructed above. First $\|A^*\|_F^2 = \sum_{l=1}^m \|A_l^*\|_2^2 = m$, since,

$$\|A_l^*\|_2^2 = \langle \psi(\boldsymbol{\gamma_l}), \psi(\boldsymbol{\gamma_l}) \rangle = g(\boldsymbol{\gamma_l}, \boldsymbol{\gamma_l}) = 1.$$

Next, we can use Lemma 3 to express the norm of B^* as $\|B^*\|_F^2 = \sum_J a_J \|g(\mathbf{x})^J\|_{\mathbf{q}}^2$ where the q_Js are the coefficients of $\Phi(\mathbf{x})$. Note that this is independent of m and given $g(\mathbf{x}), \delta$ it only depends on ε therefore we can write $\|B^*\|_F^2 = T(\varepsilon)$ where T is only a function of ε. Note that $T(\varepsilon) = \sum_J a_J \|g(\mathbf{x})^J\|_{\mathbf{q}}^2 \leq \sum_J a_J \|g^+(\mathbf{x})^J\|_{\mathbf{qu}}^2 \leq \|\sum_J a_J g^+(\mathbf{x})^J\|_{\mathbf{qu}}^2 \leq \|(\sum g_i^+(\mathbf{x}))^{c\log(1/\varepsilon)}\|_{\mathbf{qu}}^2$. By Corollary 28 this is at most $s^{O(\log(1/\varepsilon))}$.

Moving to the bound on V_{reg}, this is easy to see once we note that B^* is such that for any $l \in [m]$, for all $i \in [n]$, $B^* \sigma(C\mathbf{x}_{il}) \approx p(\mathbf{x}_{il}) = \boldsymbol{\gamma_l}$ and hence has very low intra-manifold variance.

So far we assumed the random ReLU layer to be monomials $\Phi(\mathbf{x})$ according to infinite width kernel. Now, we argue that if we use a large enough width D, then by Corollary 25 there is an orthogonal matrix U so that $\sigma(C\mathbf{x})$ is approximately $U\Phi(\mathbf{x})$. If we choose D so that $\|\sigma(C\mathbf{x}) - U\Phi(\mathbf{x})\|_2$ is at most $\varepsilon/T(\varepsilon)$ then $B^*\Phi(\mathbf{x})$ will differ from $B^*U\sigma(C\mathbf{x})$ by at most Frobenius error ε on any of the n inputs; this will result in at most additive error ε at each of the outputs in Y_i (since each row of A^* has norm at most 1. This is done by setting $D = O(\sqrt{n}T(\varepsilon)^2 \log(n/\delta)/\varepsilon^2)$.

□

Lemma 30 (Bounding the Width of the Hidden layer)**.** *Given any $\varepsilon_1, \varepsilon_2 > 0$, and A^*, B^* of Lemma 29, we can construct new A', B' with number of columns in A' (and number of rows in B') equal to $O(\log(mn)\log(1/\delta)/\varepsilon_1)$, such that*

[6] In reality there is an additional orthogonal matrix U but we can define $B_2^* = B^*U$ and subsume it in our ground truth.

1. $\mathcal{L}_{A^*,B^*}(Y,\hat{Y}) \leq \varepsilon_1$,
2. $\|A'\|_F^2 \leq m$, $\|B'\|_F^2 \leq s^{O(\log(1/\varepsilon_1))}$,
3. $\hat{V}_{mn}(B'\sigma(C.)) \leq \varepsilon_2$.

Note that now we have the small loss guarantee only on our train examples and not over any new samples from our manifolds.

Proof. Let the original width of the hidden layer (number of columns in A^*) be w. From Lemma 15, we have that randomly projecting both A^* and B^* down to $O(\log(mn)\log(1/\delta)/\varepsilon^2)$ dimensions preserves all the dot products between the normalized rows of A^* and normalized columns of B^* up to an additive error ε with probability $\geq 1-\delta$. In addition we have that $\|A_l^*\| = 1$ for all $l \in [m]$. So we can replace A^*B^* by $A'B' = (A^*R^\top)(RB^*)$ where R is the random projection matrix and get that for each input $\mathbf{x_{il}}$, $\|A'B'\sigma(C\mathbf{x_{il}}) - A^*B^*\sigma(C\mathbf{x_{il}})\|_\infty \leq \varepsilon b$ where b is the maximum norm of the rows of B^*. As an aside, we note that a similar random projection can be applied on top of the random ReLU layer $\sigma(C.)$ as well to get a random ReLU layer followed by a random projection neither of which are trained and resulting in a smaller width ReLU layer. □

Next, we recall that our objective is of the form

$$\min_{A,B} \mathcal{L}_{A,B}(Y,\hat{Y}) + \|A\|_F^2 + \|B\|_F^2 \tag{23}$$

We will argue that the nice properties we saw holding for A^*, B^* also hold for any global minima of our optimization (23). This is because of the following lemma.

Lemma 31 (Multi-Objective Optimization)**.** *Given a multi-objective minimization where we want to minimize a set of non-negative functions $O_i(\theta)$ for $i = 1,\ldots,q$ and there exists a solution θ^* such that $O_i(\theta^*) \leq OPT_i$. Then, we have that*

$$\min_\theta \sum_{i=1}^q \frac{O_i(\theta)}{OPT_i}$$

produces $\hat{\theta}$ such that for each i, $O_i(\hat{\theta}) \leq qOPT_i$ at any global minimum.

Proof. Note that at global minimum

$$\sum_{i=1}^q \frac{O_i(\theta)}{OPT_i} \leq \sum_{i=1}^q \frac{O_i(\theta^*)}{OPT_i} \leq \sum_{i=1}^q 1 = q.$$

Since O_i are non-negative functions we have $O_i(\theta) \leq qOPT_i$. □

Lemma 31 will guide our choice of regularization parameters λ_1, λ_2.

Lemma 32. *Let $\hat{A}^*, \hat{B}^*$ denote the global optimum of* (23)*. Then, for $\lambda_1 = \varepsilon_1/m, \lambda_2 = \varepsilon_1/s^{O(\log(1/\varepsilon_1))}$, we have*

$$\mathcal{L}_{\hat{A}^*,\hat{B}^*}(Y,\hat{Y}) \leq 3\varepsilon_1, \tag{24}$$

$$\|\hat{A}^*\|_F \leq 3m, \|\hat{B}^*\|_F \leq s^{O(\log(1/\varepsilon_1))}. \tag{25}$$

Proof. Recall that for ground truth A^*, B^* from Lemma 29 we have that $\mathcal{L}_{A^*,B^*}(Y,\hat{Y}) \leq \varepsilon_1$, $\|A^*\|_F^2 \leq m$ and $\|B\|_F^2 \leq s^{O(\log(1/\varepsilon_1))}$. Therefore, setting $\lambda_1 = \frac{\varepsilon_1}{m}$ and $\lambda_2 = \frac{\varepsilon_1}{s^{O(\log(1/\varepsilon_1))}}$, we get from Lemma 31 that at global minimum $\hat{A}^*, \hat{B}^*$

$$\mathcal{L}_{\hat{A}^*,\hat{B}^*}(Y,\hat{Y}) \leq 3\varepsilon_1, \tag{26}$$

$$\|\hat{A}^*\|_F \leq 3m, \|\hat{B}^*\|_F \leq 3s^{O(\log(1/\varepsilon_1))}. \tag{27}$$

□

Note that the chosen values of λ_1, λ_2 will influence the number of steps gradient descent will need to run to reach a local optimum.

Since our objective is non-convex, it is not clear how good a local optimum we reach will be. However, for our particular architecture, it turns out that every local minimum is a global minimum.

Lemma 33 (Equivalence to Nuclear Norm Regularized Convex Minimization)**.** *Let $d > \max(m, n)$. Then, for any convex objective function $O()$, in the minimization*

$$\min_{\substack{A \in \mathbb{R}^{m \times d}, \\ B \in \mathbb{R}^{d \times n}}} O(AB) + \lambda \left(\|A\|_F^2 + \|B\|_F^2 \right), \tag{P1}$$

all local minima are global minima and the above minimization is equivalent to the following convex minimization

$$\min_{\substack{A \in \mathbb{R}^{m \times d}, \\ B \in \mathbb{R}^{d \times n}}} O(AB) + 2\lambda \left(\|AB\|_* \right) \equiv \min_{W \in \mathbb{R}^{m \times n}} O(W) + 2\lambda \left(\|W\|_* \right). \tag{P2}$$

Proof. From Lemma 17, it follows that the global minimum of (P1) and (P2) have the same value. Note that the latter minimization is convex and hence any local minima is global. We now show that all local minima of (P1) are global as well even though it is potentially a non-convex objective. Let OPT denote the value of the global minimum of either objective and let A_1, B_1 be a local minima of (P1). Suppose for the sake of contradiction that $O(A_1B_1) + \lambda(\|A_1\|_F^2 + \|B_1\|_F^2) > OPT$. Then it must be the case that $\|A_1\|_F^2 + \|B_1\|_F^2 = 2\|A_1B_1\|_*$ as otherwise by Lemma 17 we will be able to improve the objective by keeping A_1B_1 a constant and reducing $\|A_1\|_F^2 + \|B_1\|_F^2$ (note that the sum of Frobenius norms given a fixed product of AB is a convex minimization problem). Therefore we have that $O(A_1B_1) + 2\lambda(\|A_1B_1\|_*) > OPT$. Since (P2) is a convex problem, this implies that for any $\varepsilon > 0$, within an ε-sized ball around $W_1 = A_1B_1$ there exists W_2 such that $O(A_1B_1) + 2\lambda(\|A_1B_1\|_*) > O(W_2) + 2\lambda(\|W_2\|_*)$. Let $W_2 = USV^\top$ be the svd of W_2 where $U \in \mathbb{R}^{m \times r}$, $S \in \mathbb{R}^{r \times r}$ and $V \in \mathbb{R}^{n \times r}$ such that S is a diagonal matrix with non-zero singular values along the diagonal, and U, V have orthonormal rows. Let $A_2 = US^{1/2}, B_2 = S^{1/2}V^\top$. Since $d > \max(m, n) > r$, we can pad A_2 with $d - r$ columns which are 0, and pad B_2 with $d - r$ rows which are 0. Even after such a padding we have that $A_2B_2 = W_2$. Then we have that $2\|A_2B_2\|_* = 2\|S^{1/2}\|_F^2 = \|A_2\|_F^2 + \|B_2\|_F^2$ and hence $O(A_2B_2) + 2\lambda(\|A_2\|_F^2 + \|B_2\|_F^2) < O(A_1B_1) + 2\lambda(\|A_1\|_F^2 + \|B_1\|_F^2)$ which is a contradiction to the statement that A_1, B_1 is a local minima of (P1) as by a continuity argument for any $\delta > 0$, we can find an appropriate ε such that $\|W_2 - W_1\|_F \leq \varepsilon$ and $\|A_2 - A_1\|_F \leq \delta$, $\|B_2 - B_1\|_F \leq \delta$. □

Corollary 34 (Generalization of Lemma 33)**.** *For any convex objective function $O()$,*

$$\min_{A,B} O(AB) + \left(\lambda_1 \|A\|_F^2 + \lambda_2 \|B\|_F^2 \right),$$

all local minima are global minima and is equivalent to the following convex objective

$$\min_{A,B} O(AB) + 2\sqrt{\lambda_1 \lambda_2} \left(\|AB\|_* \right).$$

Proof. The lemma follows by replacing A, B in the previous lemma by $\sqrt{\lambda_1/\lambda_2}A$, $\sqrt{\lambda_2/\lambda_1}B$ respectively and setting λ to $\sqrt{\lambda_1\lambda_2}$ □

Corollary 34 will imply that at any local minimum $\hat{A}, \hat{B}$ we have a small value for our weighted square loss. This is because $\mathcal{L}_{A,B}(Y, \hat{Y})$ is convex in AB. Next, we will show that an empirical variant of the GSH property holds for the representation $\hat{B}\sigma(C.)$ obtained at any local minimum. Here our approaches for linear and non-linear manifolds differ. Linear manifolds enable a more direct analysis with a plain ℓ_2-regularization. However, we need to assume certain additional conditions on the input. The result for linear manifolds acts as a warm-up to our more general result for non-linear manifolds where we have minimal assumptions but end up having to use a stronger regularizer designed to push the representation to satisfy GSH. We describe these differences in Sections D.1 and D.2.

D.1 GSH ON TRAIN DATA FOR LINEAR MANIFOLDS

Here we will show that we can train our 3-layer non-linear neural network on input data from linear manifolds, to get GSH. To get an intuitive understanding of why this is the case, we first recall that by passing an input vector $\mathbf{x}$ through a random ReLU layer, we get approximately all possible monomials of $\mathbf{x}$ and its higher tensor powers (Corollary 25). Now, we will show that by passing in a dummy constant as part of the input, the regularization on the weights A and B enforces that weights corresponding to certain monomials of $\mathbf{x}$ are zero at any minima. These weights being zero will imply Property A of the hashing property. The second part of the hashing property will follow due to a similar reasoning as in Section D.2. With this high level intuition in mind, we proceed with the formal proof.

A linear manifold with a latent vector $\boldsymbol{\gamma}$ can be represented by the set $\{\mathbf{x} = P\boldsymbol{\theta} + Q\boldsymbol{\gamma}\}$ for some matrices P and Q. Moreover, without a significant loss of generality we can assume that $\boldsymbol{\gamma}$ is such that $Q\boldsymbol{\gamma} \perp P\boldsymbol{\theta}$ for all $\boldsymbol{\theta} \in \mathcal{S}^{k-1}$ (as otherwise, we can project $Q\boldsymbol{\gamma}$ onto the subspace perpendicular to $P\boldsymbol{\theta}$). The objective function we minimize is (23).

We begin the proof by first performing a transformation on the input that will simplify the presentation.

Lemma 35. *Given a point $\mathbf{x} = P\boldsymbol{\theta} + Q\boldsymbol{\gamma}$ where $P\boldsymbol{\theta} \perp Q\boldsymbol{\gamma}$, there exists an orthogonal matrix U_2*

$$\sigma(C\mathbf{x}) = U_2\phi(\boldsymbol{\gamma}', \boldsymbol{\theta}') + \delta,$$

where $\boldsymbol{\gamma}' = RQ\boldsymbol{\gamma}$ and $\boldsymbol{\theta}' = RP\boldsymbol{\theta}$ and U and $\phi(.)$ are as defined in Corollary 25.

Proof. Since $P\boldsymbol{\theta} \perp Q\boldsymbol{\gamma}$, a rotation of the bases transforms $Q\boldsymbol{\gamma}$ to a vector with non-zero entries only in the first $d-k$ coordinates, and $P\boldsymbol{\theta}$ to lie in a subspace which contains vectors with non-zero entries only in the last k coordinates. This is made feasible since the rank of the space spanned by $\boldsymbol{\theta}$ is $\leq k$. Denote the vectors obtained after these transformations by $\boldsymbol{\gamma}'$ and $\boldsymbol{\theta}'$. We drop the 0 entries to get $\boldsymbol{\gamma}' \in \mathbb{R}^{d-k}$ and $\boldsymbol{\theta}' \in \mathbb{R}^k$. Therefore, $\mathbf{x} = R_2(\boldsymbol{\gamma}', \boldsymbol{\theta}')$ for some rotation matrix R_2. Note that rotation matrices are orthogonal. Now $\sigma(C\mathbf{x}) = \sigma(CR_2(\boldsymbol{\gamma}', \boldsymbol{\theta}'))$. CR is also a random matrix distributed according to $\mathcal{N}\left(0, \frac{I}{D}\right)$ and hence Corollary 25 applies to it as well giving us the statement of the Lemma. □

In light of Lemma 35, we can assume that our neural net gets as input $\tilde{\mathbf{x}} = (\boldsymbol{\gamma}', \boldsymbol{\theta}')$ where $\boldsymbol{\gamma}' \in \mathbb{R}^{(d-k)}$ and $\boldsymbol{\theta}' \in \mathbb{R}^k$ without loss of generality as after passing through the random ReLU layer all that differs between the two views is the orthogonal matrix U which is applied to $\phi(\mathbf{x})$. In addition to the constant $1/\sqrt{2}$ appending to our input originally, we append a constant $\frac{\sqrt{k(d+1)}}{\sqrt{(d+k)}}$ as well to $\tilde{\mathbf{x}}$ before passing it to our neural network as this will help us argue GSH.

D.1.1 PROPERTY (A) OF GSH FOR LINEAR MANIFOLDS

A key part of our argument for why we can get neural nets to behave as hash functions over manifold data is the observation that at the output layer having a small variance over points from the same manifold benefits the primary component of the loss. The following lemma formalizes the above intuition focusing on a single manifold. Note that this result holds for non-linear manifolds too.

Lemma 36 (Centering)**.** *Let M be one of the train manifolds with associated latent vector $\boldsymbol{\gamma}$. For each $\mathbf{x} \sim \mathcal{D}(M)$, replacing $\hat{\mathbf{y}}$ by $\hat{\mathbf{y}}' = \mathbb{E}_n[\hat{\mathbf{y}}|\boldsymbol{\gamma}] = \frac{1}{n}\sum_{i=1}^{n} \hat{\mathbf{y}}_\mathbf{i}$ will reduce the (weighted) square loss term corresponding to M*

$$\mathcal{L}(Y, \hat{Y}) = \frac{1}{n}\sum_{i=1}^{n}\sum_{l=1}^{m} \|\mathbf{w_l} \odot (\hat{\mathbf{y}}_\mathbf{il} - \mathbf{y_{il}})\|_2^2$$

by at least $\hat{V}_{mn}(\hat{\mathbf{y}})$.

Proof. We start by focusing on a single manifold with latent vector $\boldsymbol{\gamma}$. We drop the conditioning on $\boldsymbol{\gamma}$ to simplify the proof. Note that if there is no weighting of different coordinates of $\mathbf{y}$ according to

$\mathbf{w_1}$, then

$$\begin{aligned}\mathop{\mathbb{E}}_{n}[\|\mathbf{y}-\hat{\mathbf{y}}\|_2^2] &= \mathop{\mathbb{E}}_{n}[\|\mathbf{y}-\hat{\mathbf{y}}'+\hat{\mathbf{y}}'-\hat{\mathbf{y}}\|_2^2] = \|\mathbf{y}-\hat{\mathbf{y}}'\|_2^2 + \mathop{\mathbb{E}}_{n}[\|\hat{\mathbf{y}}'-\hat{\mathbf{y}}\|_2^2] + 2\mathop{\mathbb{E}}_{n}[(\mathbf{y}-\hat{\mathbf{y}}')^\top(\hat{\mathbf{y}}'-\hat{\mathbf{y}})] \\ &= \|\mathbf{y}-\hat{\mathbf{y}}'\|_2^2 + \mathop{\mathbb{E}}_{n}[\|\hat{\mathbf{y}}'-\hat{\mathbf{y}}\|_2^2] + 2(\mathbf{y}-\hat{\mathbf{y}}')^\top \mathop{\mathbb{E}}_{n}[\hat{\mathbf{y}}'-\hat{\mathbf{y}}] \\ &= \|\mathbf{y}-\hat{\mathbf{y}}'\|_2^2 + \hat{V}_n(\hat{\mathbf{y}}|\boldsymbol{\gamma}) + 0. \end{aligned} \tag{28}$$

So the value of $(1/m)\,\mathbb{E}_n[\|\mathbf{y}-\hat{\mathbf{y}}\|_2^2]$ reduces by at least $\hat{V}_n(\hat{\mathbf{y}}|\boldsymbol{\gamma}))/m$ upon replacing $\hat{\mathbf{y}}$ by its average value per manifold $\hat{\mathbf{y}}'$. This holds even when there is weighting according to the $\mathbf{w}$ matrix as it only depends on $\boldsymbol{\gamma}$ and doesn't vary based on $\boldsymbol{\theta}$.

$$\begin{aligned}\mathop{\mathbb{E}}_{n}[\|\mathbf{w}_{\boldsymbol{\gamma}}\odot(\mathbf{y}-\hat{\mathbf{y}})\|_2^2] &= \mathop{\mathbb{E}}_{n}[\|\mathbf{w}_{\boldsymbol{\gamma}}\odot(\mathbf{y}-\hat{\mathbf{y}}'+\hat{\mathbf{y}}'-\hat{\mathbf{y}})\|_2^2] \\ &= \|\mathbf{w}_{\boldsymbol{\gamma}}\odot(\mathbf{y}-\hat{\mathbf{y}}')\|_2^2 + \mathop{\mathbb{E}}_{n}[\|\mathbf{w}_{\boldsymbol{\gamma}}\odot(\hat{\mathbf{y}}'-\hat{\mathbf{y}})\|_2^2] + 2\mathop{\mathbb{E}}_{n}[(\mathbf{w}_{\boldsymbol{\gamma}}\odot(\mathbf{y}-\hat{\mathbf{y}}'))^\top(\mathbf{w}_{\boldsymbol{\gamma}}\odot(\hat{\mathbf{y}}'-\hat{\mathbf{y}}))] \\ &= \|\mathbf{w}_{\boldsymbol{\gamma}}\odot(\mathbf{y}-\hat{\mathbf{y}}')\|_2^2 + \mathop{\mathbb{E}}_{n}[\|\mathbf{w}_{\boldsymbol{\gamma}}\odot(\hat{\mathbf{y}}'-\hat{\mathbf{y}})\|_2^2] + (\mathbf{w}_{\boldsymbol{\gamma}}\odot(\mathbf{y}-\hat{\mathbf{y}}'))^\top\left(\mathbf{w}_{\boldsymbol{\gamma}}\odot\mathop{\mathbb{E}}_{n}[\hat{\mathbf{y}}'-\hat{\mathbf{y}}|\boldsymbol{\gamma}]\right) \\ &= \|\mathbf{w}_{\boldsymbol{\gamma}}\odot(\mathbf{y}-\hat{\mathbf{y}}')\|_2^2] + \mathop{\mathbb{E}}_{n}[\|\mathbf{w}_{\boldsymbol{\gamma}}\odot(\hat{\mathbf{y}}'-\hat{\mathbf{y}})\|_2^2] + 0.\end{aligned}$$

Thus even in the weighted case the weighted square loss gets reduced by at least $\mathbb{E}_n[\|\mathbf{w}_{\boldsymbol{\gamma}}\odot(\hat{\mathbf{y}}'-\hat{\mathbf{y}})\|_2^2]$. But since $\mathbf{w}_{\boldsymbol{\gamma}}$ is at least $1/\sqrt{2(m-1)}$ per coordinate, this is at least $\hat{V}_n(\hat{\mathbf{y}}|\boldsymbol{\gamma})/(2(m-1))$. Summing up this reduction over all the m manifolds, we get the lemma. □

The following lemma will show that it is in fact beneficial to have a zero variance at the representation layer itself rather than just at the output layer. This also holds generally across linear and non-linear manifolds.

Lemma 37. *$\hat{V}_{mn}(\hat{\mathbf{y}}) = 0$ if and only if the variance of the representation layer $\hat{V}_{mn}(\mathbf{r}) = 0$ where $\mathbf{r} = B\sigma(C\mathbf{x})$. Further $\hat{V}_{mn}(\hat{\mathbf{y}}) \geq (\lambda_2/\lambda_1)(\hat{V}_{mn}(\mathbf{r}))^2/4$ where λ_1, λ_2 are the regularization weights.*

Proof. Recall that for a non-square matrix $W = USV^\top$, we define the square root as $W^{1/2} = US^{1/2}V^\top$. From Corollary 20, we have that at local minima of (4), if $\hat{\mathbf{y}} = W\mathbf{z}$ then without loss of generality $\mathbf{r} = W^{1/2}\mathbf{z}$ (upto orthonormal rotation and scaling). Let $\mathbf{z}' = \mathbf{z} - \mathbb{E}_n[\mathbf{z}]$ where the mean value of z per manifold has already been subtracted. Let Z' denote the matrix of all such $\mathbf{z}'$ scaled by $1/\sqrt{n}$. Since $\hat{\mathbf{y}}, \mathbf{r}$ are linear transforms of $\mathbf{z}$, it is not hard to see that $\hat{V}_{mn}(\hat{\mathbf{y}}) = \|WZ'\|_F^2$ and similarly $\hat{V}_{mn}(\mathbf{r}) = \|W^{1/2}Z'\|_F^2$. Now, if $\|W^{1/2}Z'\|_F = 0$, then $\|WZ'\|_F = 0$ clearly. To see the other direction, we first observe that multiplying Z' by a matrix W is the same as taking dot products of the columns of Z' with the right singular vectors of W and scaling the result by the singular values of W. Now, the right singular vectors of W and $W^{1/2}$ are the same and the singular value of $W = 0$ iff the corresponding singular value of $W^{1/2}$ is 0 as well. Therefore, if $\|WZ'\|_F = 0$, then so is $\|W^{1/2}Z'\|_F$.

Furthermore, the singular values of $W^{1/2}$ are square roots of the singular values of W. So $\|W^{1/2}Z'\|_F$ can be non-zero only if and only if Z' has component along a singular vector of $W^{1/2}$ with non-zero singular value and the same must be true for $\|WZ'\|_F$ as well. Note that since W has m rows there are at most m singular values $\sigma_1, \ldots, \sigma_m$. Let $c_1, \ldots, c_m$ be the total norm squared of Z' along the right singular vectors, that is if $W = UDV^\top$, then $c_i = \|V_{i,*}^\top Z'\|_2^2$. Since for any $\mathbf{z}$, $\|z\|_2 \leq \|C\|_2\alpha$, same must be true for $\mathbf{z}'$. Hence, $\sum c_i \leq \|C\|_2^2\alpha^2$. Now, $\hat{V}_{mn}(\hat{\mathbf{y}}) = \sum \sigma_i^2 c_i$ and $\hat{V}_{mn}(\mathbf{r}) = \sum \sigma_i c_i$. Now in the latter, the sum from those singular values that are at most $\hat{V}_{mn}(\mathbf{r})/2$ is at most $\hat{V}_{mn}(\mathbf{r})/2$ and so the rest must be coming from singular values larger than $\hat{V}_{mn}(\mathbf{r})/2$. Since the singular vectors are getting squared we have $\hat{V}_{mn}(\hat{\mathbf{y}}) \geq (\hat{V}_{mn}(\mathbf{r})/2)(\hat{V}_{mn}(\mathbf{r})/2) = (\hat{V}_{mn}(\mathbf{r}))^2/4$. If the weights of $\|A\|_F^2, \|B\|_F^2$ are λ_1, λ_2 then $B = \sqrt{\frac{\lambda_1}{\lambda_2}}W^{1/2}$ and the statement of the lemma follows. □

Next we show that if the intra-manifold variance of the representation $\mathbf{r} = B\sigma(C\mathbf{x})$ is large for a certain weight matrix B then replacing $\mathbf{r}$ by its mean value per manifold leads to a reduction in the intra-manifold variance down to a small value. Moreover, this can be achieved by using a B' such that $\|B'\|_F \leq \|B\|_F$. The main idea is more easily exposited by first assuming we have an infinite width random ReLU layer. Hence, we first state the following lemma which shows that we can push the intra-manifold variance of the representation all the way down to 0 if $D \to \infty$.

Lemma 38. *For $D \to \infty$, given as input $\tilde{\mathbf{x}}' = (\boldsymbol{\gamma}', \boldsymbol{\theta}', \frac{\sqrt{k(d+1)}}{\sqrt{(d+k)}})$, and given a B we can transform it to B' with no greater Frobenius norm so that $\hat{V}_{mn}(B'\mathbf{z}) = 0$.*

Proof. First let us assume that instead of the constant $\frac{\sqrt{k(d+1)}}{\sqrt{(d+k)}}$ being appended to the input, we have k 1s appended. We will later argue that the features computed by the ReLU layer are equivalent for both these cases. Let $\sigma(C\mathbf{x}) = U_2\phi(\boldsymbol{\gamma}', \boldsymbol{\theta}') + \boldsymbol{\delta}$. When $D \to \infty$, we have from Corollary 25 that $\boldsymbol{\delta} \to 0$. Consider the matrix $B_2 = BU_2$. Then B_2 can be viewed as a linear mapping from different monomials in the representation computed by the random ReLU layer to a new representation space. Crucially, since $\phi(\boldsymbol{\gamma}', \boldsymbol{\theta}')$ comprises of monomials in $\boldsymbol{\gamma}', \boldsymbol{\theta}'$, we can view each column of B_2 as the set of weights corresponding to a particular monomial in $\tilde{\mathbf{x}}$. Let $M(\boldsymbol{\gamma}', \boldsymbol{\theta}')$ be one such monomial. Suppose the intra-manifold variance of $\mathbf{r}$ is larger than 0, then B_2 matrix must have non-zero weight on nodes which correspond to monomials $M(\boldsymbol{\gamma}', \boldsymbol{\theta}')$ that depend on $\boldsymbol{\theta}'$. By Lemmas 36 and 37, replacing the terms depending on $\boldsymbol{\theta}'$ by their expected value over $\boldsymbol{\theta}'$ reduces $\hat{V}_{mn}(\hat{\mathbf{y}})$ and consequently also the square loss term. Since we have the $\mathbf{1}_k$ vector concatenated to $\tilde{x}$, for each monomial $M(\boldsymbol{\gamma}', \boldsymbol{\theta}')$ there is a unique corresponding monomial $M(\boldsymbol{\gamma}', \mathbf{1})$ with the same coefficient as $M(\boldsymbol{\gamma}', \boldsymbol{\theta}')$. This is because whatever combination of coordinates of $\boldsymbol{\gamma}'$ and $\boldsymbol{\theta}'$ and their powers are chosen in $M(\boldsymbol{\gamma}', \boldsymbol{\theta}')$ we can choose the same combination of coordinates and powers to get $M(\boldsymbol{\gamma}', \mathbf{1}_k)$ where we have replaced $\boldsymbol{\theta}'$ with $\mathbf{1}_k$. And note that since we have assumed $\boldsymbol{\gamma}', \boldsymbol{\theta}'$ vectors to lie within the unit sphere, $\mathbb{E}_n[M(\boldsymbol{\gamma}', \boldsymbol{\theta}')] = c'M(\boldsymbol{\gamma}', \mathbf{1})$ where $c' \leq 1$. This implies that shifting the weights of B_2 from terms corresponding to the monomials which depend on $\boldsymbol{\theta}'$ to those corresponding to monomials which have no $\boldsymbol{\theta}'$ dependence should strictly decrease the square loss. Moreover, this shift ensures that $\|B_2\|_F$ is not increased. Since $B = B_2U_2^\top$, we have that $\|B\|_F = \|B_2\|_F$ and hence $\|B\|_F$ does not increase as well. A has remained unmodified throughout this process and hence we have managed to strictly decrease the loss by decreasing $\hat{V}_{mn}(B\mathbf{z})$ to 0 at the same time.

Now we argue that, appending k 1s to our input produces the same output as appending a single scalar $\frac{\sqrt{k(d+1)}}{\sqrt{(d+k)}}$ where the original dimension of input be d. Recall that the weight matrix for the ReLU layer is randomly initialized with each row being drawn from $\mathcal{N}(0, I/D)$. Consider the output being computed at any single node after the ReLU. In the first case, the contribution of the k 1s to the output is $\sum_{i=1}^{k} c_i$ where each $c_i \sim \mathcal{N}(0, 1/(D(d+k)))$. This is equivalent to a single $c \sim \mathcal{N}(0, k/(D(d+k)))$. So by appending a constant of value $\frac{\sqrt{k(d+1)}}{\sqrt{(d+k)}}$ the same effect will be achieved. □

Next we show that the insights of Lemma 38 continue to hold approximately for a finite D as long as it is large enough.

Lemma 39. *For $D \geq O(\sqrt{nm}\log(mn/\delta)/\varepsilon)$, given as input $\tilde{\mathbf{x}}' = (\boldsymbol{\gamma}', \boldsymbol{\theta}', \frac{\sqrt{k(d+1)}}{\sqrt{(d+k)}})$, and given a B we can transform it to B' with no greater Frobenius norm so that $\hat{V}_{mn}(B'\mathbf{z}) \leq 4\varepsilon$.*

Proof. Let $\sigma(C\mathbf{x}) = U_2\phi(\boldsymbol{\gamma}', \boldsymbol{\theta}') + \boldsymbol{\delta}$. When $D \geq O(\sqrt{nm}\log(mn/\delta)/\varepsilon)$, we have from Corollary 25 that $\|\boldsymbol{\delta}\|_2 \leq 2\sqrt{\varepsilon}$. This will imply that the intra-manifold variance of $\mathbf{r}$ when B_2 has zero weight on nodes with a $\boldsymbol{\theta}'$ dependence is at most $\|2\boldsymbol{\delta}\|_2^2/4 = (4\sqrt{\varepsilon})^2/4 = 4\varepsilon$. The rest of the argument proceeds similar to before. Now at each output node of the ReLU layer, we compute a monomial $M(\boldsymbol{\gamma}', \boldsymbol{\theta}') + \delta_i$ where δ_i is the i^{th} coordinate of norm-bounded noise. If $\hat{V}_{mn}(B\sigma(C.)) > 4\varepsilon$, then shifting the weights of B in a similar manner as we did in Lemma 38 will yield a B' such that $\hat{V}_{mn}(B\sigma(C.)) \leq 4\varepsilon$ while maintaining $\|B\|_F = \|B'\|_F$. This ultimately leads to a decrease in the overall objective value. □

Lemma 39 gives the following.

Lemma 40. *At any minima of the objective function* (23) *over points taken from the distribution of* $\boldsymbol{\gamma}, \boldsymbol{\theta}$, $\hat{V}_{mn}(\mathbf{r}) \leq 4\varepsilon$.

Proof. Assume that at some minima $\hat{V}_{mn}(\mathbf{r}) > 4\varepsilon$. Then by Lemma 37, $\hat{V}_{mn}(\hat{\mathbf{y}}) \geq (\lambda_2/\lambda_1)4\varepsilon^2 = 4m\varepsilon^2/s^{O(\log(1/\varepsilon))}$. Now by replacing B by B' as described in Lemma 39, $\hat{V}_{mn}(\mathbf{r})$ is pushed to a value smaller than 4ε which implies that the output $\hat{\mathbf{y}}$ will be replaced by its approximate mean $\hat{\mathbf{y}}'$ per manifold which by Lemma 36 reduces the weighted square loss. A remains unchanged and B's Frobenius norm has not increased. So the value of the minimization objective overall has reduced which is a contradiction to Lemma 33 which states that all minima are global in our setting. □

Lemma 40 implies that property (A) holds on the train examples from the train manifolds.

D.1.2 Property (B) of GSH for Linear Manifolds

Next we prove a bunch of Lemmas for showing property (B) of GSH for linear manifolds. In this section, without loss of generality we will assume that m is even. If it is not, we drop the samples from the m^{th} manifold and set the new value of m to be $m-1$. We will use the fact that our train loss is small. The following Lemmas will argue that when the train loss is small, the average of the inter-manifold representation distance over all pairs of our train manifolds is small.

Lemma 41. *Let* $\mathbf{a} \in \mathbb{R}^T$ *such that* $\|\mathbf{a}\|_2 \leq \delta$. *Let* $\mathbf{b_1}, \mathbf{b_2} \in \mathbb{R}^T$ *be such that*

$$\frac{1}{2}\left(\mathbf{a}^\top \mathbf{b_1}\right)^2 + \frac{1}{2}\left(1 - \mathbf{a}^\top \mathbf{b_2}\right)^2 \leq \varepsilon.$$

Then

$$\|\mathbf{b_1} - \mathbf{b_2}\|_2^2 \geq \frac{1 - 4\sqrt{\varepsilon}}{\delta^2}.$$

Proof. Let $(\mathbf{a}^\top \mathbf{b_1})^2 = 2\varepsilon_1$ and let $\varepsilon_2 = \varepsilon - \varepsilon_1$. Then,

$$\left|\mathbf{a}^\top \mathbf{b_1}\right| \leq \sqrt{2\varepsilon_1} \text{ and } \left|1 - \mathbf{a}^\top \mathbf{b_2}\right| \leq \sqrt{2\varepsilon_2} \tag{29}$$

$$\implies |\mathbf{a}^\top(\mathbf{b_2} - \mathbf{b_1})| \geq 1 - \sqrt{2\varepsilon_1} - \sqrt{2\varepsilon_2} \geq 1 - 2\sqrt{\varepsilon} \tag{30}$$

$$\implies \|\mathbf{a}\|_2 \|\mathbf{b_2} - \mathbf{b_1}\|_2 \geq 1 - \sqrt{2\varepsilon} \implies \|\mathbf{b_2} - \mathbf{b_1}\|_2 \geq \frac{1 - 2\sqrt{\varepsilon}}{\delta} \tag{31}$$

$$\implies \|\mathbf{b_2} - \mathbf{b_1}\|_2^2 \geq \frac{1 + 4\varepsilon - 4\sqrt{\varepsilon}}{\delta^2} \geq \frac{1 - 4\sqrt{\varepsilon}}{\delta^2}. \tag{32}$$

where we used that $\sqrt{2(a+b)} \geq \sqrt{a} + \sqrt{b}$. □

Lemma 42. *For* $l \in [m]$, *let*

$$H(l) = \frac{1}{n(m-1)} \sum_{j \neq l} \sum_{i=1}^{n} \|B\sigma(C\mathbf{x_{il}}) - B\sigma(C\mathbf{x_{ij}})\|_2^2, \tag{33}$$

$$\varepsilon_l = \frac{1}{n(m-1)} \sum_{i=1}^{n} \left[\sum_{j \neq l} \varepsilon_{ijl} \right], \tag{34}$$

$$\textit{where } \varepsilon_{ijl} = \frac{1}{2}\left(A_l B\sigma(C\mathbf{x_{ij}})\right)^2 + \frac{1}{2}\left(1 - A_l B\sigma(C\mathbf{x_{il}})\right)^2. \tag{35}$$

We have,

$$H(l) \geq \frac{1 - 4\sqrt{\varepsilon_l}}{\|A_l\|_2^2}.$$

Proof. From Lemma 41 we have

$$\|B\sigma(C\mathbf{x_{il}}) - B\sigma(C\mathbf{x_{ij}})\|_2^2 \geq \frac{1 - 4\sqrt{\varepsilon_{ijl}}}{\|A_l\|_2^2} \tag{36}$$

$$\implies \frac{1}{n(m-1)} \sum_{i=1}^{n} \sum_{j \neq l} \|B\sigma(C\mathbf{x_{il}}) - B\sigma(C\mathbf{x_{ij}})\|_2^2 \geq \frac{1}{n(m-1)} \frac{n(m-1) - 4n(m-1)\sqrt{\varepsilon_l}}{\|A_l\|_2^2} \tag{37}$$

$$= \frac{1 - 4\sqrt{\varepsilon_l}}{\|A_l\|_2^2}, \tag{38}$$

where we have used that $\sum_{k=1}^{c} \sqrt{a_k} \leq \sqrt{c} \cdot \sqrt{\sum_{k=1}^{n} a_k}$. □

Lemma 43 (Small Weighted Square Loss Implies Distant Representations)**.** *For $l \in [m]$, let*

$$H(l) = \frac{1}{n(m-1)} \sum_{j \neq l} \sum_{i=1}^{n} \|B\sigma(C\mathbf{x_{il}}) - B\sigma(C\mathbf{x_{ij}})\|_2^2 \tag{39}$$

$$\varepsilon = \mathcal{L}_{A,B}(Y, \hat{Y}), \tag{40}$$

Then,

$$\frac{1}{m} \sum_{l=1}^{m} H(l) \geq \frac{m(1 - O(\sqrt{\varepsilon}))}{O(\|A\|_F^2)}.$$

Proof. First note that $\varepsilon = \sum_{l=1}^{m} \varepsilon_l / m$ and let $\delta = \sum_{l=1}^{m} \|A_l\|_2^2 / m$. From Markov's inequality we have that there exists $S_1 \subseteq [m]$ such that $|S_1| = \lceil 9m/10 \rceil$ and $\forall l \in S_1, \varepsilon_l \leq 10\varepsilon$. Similarly there exists $S_2 \subseteq [m]$, $|S_2| = \lceil 9m/10 \rceil$ such that $\forall l \in S_2, \|A_l\|_2^2 \leq 10\delta$. Note that $|S_1 \cap S_2| \geq \lceil 8m/10 \rceil$. We have,

$$\frac{1}{m} \sum_{l=1}^{m} H(l) \geq \frac{1}{m} \sum_{l \in |S_1 \cap S_2|} H(l) \geq \frac{8}{10} \cdot \frac{1 - 4\sqrt{10\varepsilon}}{10\delta} = \frac{m(1 - O(\sqrt{\varepsilon}))}{O(\|A\|_F^2)}, \tag{41}$$

where we used Lemma 42. □

Now since at any local minimum $\mathcal{L}_{\hat{A},\hat{B}}(Y, \hat{Y}) \leq \varepsilon$ and $\|\hat{A}\|_F^2 \leq 3m$, Lemma 43 gives property (B) on the train data.

D.2 GSH on Train Data for Non-Linear Manifolds With Intra-Class Variance Regularization

For non-linear manifolds, the argument we had in Section D.1 does not go through as is. This is because we no longer have as nice a mapping from monomials of $\mathbf{x}$ to associated monomials of similar degree in $\boldsymbol{\gamma}, \boldsymbol{\theta}$ as we had before. In particular, our argument for Lemma 39 breaks down. Instead, we show a more general result over non-linear manifolds in this section via the means of a different form of regularizer than before.

Recall that we add to our objective the following variance regularization term

$$V_{\text{reg}}(B\sigma(C\cdot)) = \frac{n}{n-1} \hat{V}_{mn}(B\sigma(C\cdot))$$

The final objective we minimize is,

$$\mathcal{L}_{A,B}(Y, \hat{Y}) + \lambda_1 \|A\|_F^2 + \lambda_2 \left(\|B\|_F^2 + V_{\text{reg}}(B\sigma(C)) \right) \tag{42}$$

We prove Theorem 4 via a series of Lemmas which follow. We begin by showing that for this new minimization objective, the ground truth matrices A^*, B^* from Lemma 29 still give good properties.

Lemma 44 (Good Ground Truth for Non-Linear Manifolds)**.** *Given the ground truth weight matrices A^*, B^* of Lemma 29, we have that*

$$V_{reg}(B^*\sigma(C.)) \leq 2\varepsilon_2.$$

Proof. This follows immediately from Lemma 29, point 4 since $V_{\text{reg}}(B^*\sigma(C.)) = \frac{n}{n-1}\hat{V}_{mn}(B^*\sigma(C.))$. □

Next, we show that, remarkably, even for our new objective the property that all local minima are global still holds.

Lemma 45. *Consider* (42)*. Every local minimum is a global minimum.*

Proof. We will map our minimization to an appropriate form whereafter we can apply Lemma 33 to argue that it is equivalent to a weighted nuclear norm minimization in AB which is convex in AB. Let $Z'' = (I, Z')$ where we stacked the identity's columns at the front of Z'. Note that Z'' is full row rank. Consider the SVD of $Z'' = USV^\top$ and consider the truncated form $Z''_{trunc} = US_{trunc}$. Z''_{trunc} is a square matrix which is invertible and

$$\|B\|_F^2 + \|BZ'\|_F^2 = \|BZ''\|_F^2 = \|BZ''_{trunc}\|_F^2. \tag{43}$$

By letting $B'' = BZ''_{trunc}$, we can re-write (42) in the following form now.

$$\min_{A,B''} \sum_{l=1}^{m} \frac{1}{m}\|W_l \odot (Y_l - AB''M_l)\|_F^2 + \lambda_1\|A\|_F^2 + \lambda_2\|B''\|_F^2, \tag{44}$$

where $M_l = S_{trunc}^{-1}U^\top Z_l$ which is a minimization of a convex function of AB'' with Frobenius norm regularization which by the application of Lemma 33 gives us the desired mapping to a convex function minimization with nuclear norm regularization which is a convex objective and gradient descent on this objective will achieve the global minimum which we can argue is also what is achieved by gradient descent on our objective. □

Lemma 46. *We get that at any local minimum $\hat{A}, \hat{B}$ of* (42) *for $\lambda_1 = \varepsilon_1/m$, $\lambda_2 = \varepsilon_1/s^{O(\log(1/\varepsilon_1))}$, we have*

$$\mathcal{L}_{\hat{A},\hat{B}}(Y, \hat{Y}) \leq 4\varepsilon_1, \tag{45}$$

$$\|\hat{A}\|_F \leq 4m, \|\hat{B}\|_F \leq 4s^{O(\log(1/\varepsilon_1))}, \tag{46}$$

$$V_{reg}(B^*\sigma(C.)) \leq 8\varepsilon_2. \tag{47}$$

Proof. Following a line of argument similar to the proof of Lemma 32 we get that at global minimum the bounds stated in the Lemma are satisfied (by using Lemma 31). Lemma 45 gives us that all local minima are global and hence the statement of the current lemma follows. □

Next we show that that each part of GSH holds on the train data.

D.2.1 Property (A) of GSH

Since at any local minimum, we have that $V_{\text{reg}}(\hat{B}\sigma(C.)) = \frac{n}{n-1}\hat{V}_{mn}(\hat{B}\sigma(C.))$ and $V_{\text{reg}}(\hat{B}\sigma(C.)) \leq 8\varepsilon_2$ from Lemma 44 we get that $\hat{V}_{mn}(\hat{B}\sigma(C.)) \leq 8\varepsilon_2$ as well immediately giving property (A) of the GSH.

D.2.2 Property (B) of GSH

Recall the argument for showing property B for linear manifolds from Section D.1. Lemmas 41-43 imply that the average inter-manifold representation distance is larger than a constant as long as $\mathcal{L}_{\hat{A},\hat{B}}(Y, \hat{Y})$ is small and $\|\hat{A}\|_F$ is small. These two properties still hold in the current setting for non-linear manifolds. Hence we immediately get that property (B) holds on the train data for non-linear manifolds as well.

E GENERALIZATION BOUNDS USING RADEMACHER COMPLEXITY

In this section, we present population variants for bounds on empirical quantities that we saw in Section D. Since the architectures for linear and non-linear manifolds are the same, the results in this section will apply to both. We rely on the technique of uniform convergence bounds which are shown via Rademacher complexity.

A recurring general function class whose Rademacher complexity we will repeatedly use is the following

$$\mathcal{F} = \left\{ f(\mathbf{x}) = \|P\mathbf{x} + \mathbf{b}\|_2^2 \middle| \|\mathbf{x}\|_2 \leq \alpha, \|P\|_F \leq \beta, \|\mathbf{b}\|_2 \leq b \right\}.$$

We present a Rademacher complexity bound for this class.

Lemma 47. *Given the above function class $\mathcal{F}$, we have*

$$\mathcal{R}_n(\mathcal{F}) \leq \frac{2b\beta\alpha + \beta^2\alpha^2}{\sqrt{n}}.$$

Proof. First, $\|P\mathbf{x} + \mathbf{b}\|_2^2 = \|P\mathbf{x}\|_2^2 + \|\mathbf{b}\|_2^2 + 2\mathbf{b}^\top P\mathbf{x}$. Since $\sup_\theta(f_\theta(\mathbf{x}) + g_\theta(\mathbf{x})) \leq \sup_\theta f_\theta(\mathbf{x}) + \sup_\theta g_\theta(\mathbf{x})$ we have

$$\mathcal{R}_n(\mathcal{F}) = \frac{1}{n}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{f\in\mathcal{F}} \sum_{i=1}^n \xi_i f(\mathbf{x_i})\right] \tag{48}$$

$$\leq \underbrace{\frac{1}{n}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{P:\|P\|_F\leq\beta} \sum_{i=1}^n \xi_i \|P\mathbf{x_i}\|_2^2\right]}_{(A)} + \underbrace{\frac{1}{n}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{\mathbf{b}:\|\mathbf{b}\|_2\leq b} \sum_{i=1}^n \xi_i \|\mathbf{b}\|_2^2\right]}_{(B)} + \underbrace{\frac{1}{n}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{\substack{P,\mathbf{b}:\|P\|_F\leq\beta,\\ \|\mathbf{b}\|_2\leq b}} \sum_{i=1}^n \xi_i 2\mathbf{b}^\top P\mathbf{x_i}\right]}_{(C)}. \tag{49}$$

We bound each term separately. (B) is clearly 0. (C) is the Rademacher complexity of a class of linear functions of $\mathbf{x}$ which is known to be bounded by $\|\mathbf{b}^\top P\|_2\alpha/\sqrt{n}$ (Lemma 26.10 of (Shalev-Shwartz & Ben-David, 2014)) which can be bounded by $2b\beta\alpha/\sqrt{n}$. It remains to bound (A). Here we use that,

$$\|P\mathbf{x}\|_2^2 = P^\top P \odot \mathbf{x}\mathbf{x}^\top. \tag{50}$$

Moreover, we have $\|P^\top P\|_* = \|P\|_F^2$. Now,

$$
\begin{aligned}
(A) &= \frac{1}{n}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{\|P\|_F\le\beta}\sum_{i=1}^{n}\xi_i\|P\mathbf{x_i}\|_2^2\right] \\
&= \frac{1}{n}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{\|P\|_F\le\beta}\sum_{i=1}^{n}\xi_i\left(P^\top P\odot\mathbf{x_i}\mathbf{x_i}^\top\right)\right] &(51)\\
&\le \frac{1}{n}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{\|W\|_*\le\beta^2}\sum_{i=1}^{n}\xi_i\left(W\odot\mathbf{x_i}\mathbf{x_i}^\top\right)\right] &(52)\\
&= \frac{1}{n}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{\|W\|_*\le\beta^2}\left(W\odot\sum_{i=1}^{n}\xi_i\mathbf{x_i}\mathbf{x_i}^\top\right)\right] \\
&= \frac{\beta^2}{n}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\left\|\sum_{i=1}^{n}\xi_i\mathbf{x_i}\mathbf{x_i}^\top\right\|_2\right] &(53)\\
&\le \frac{\beta^2}{n}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\left\|\sum_{i=1}^{n}\xi_i\mathbf{x_i}\mathbf{x_i}^\top\right\|_F\right] &(54)\\
&\le \frac{\beta^2}{n}\sqrt{\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\left\|\sum_{i=1}^{n}\xi_i\mathbf{x_i}\mathbf{x_i}^\top\right\|_F^2\right]} &(55)\\
&= \frac{\beta^2}{n}\sqrt{\sum_{i=1}^{n}\|\mathbf{x_i}\mathbf{x_i}^\top\|_F^2} &(56)\\
&= \frac{\beta^2}{n}\sqrt{\sum_{i=1}^{n}\|\mathbf{x_i}\|_2^4} \le \frac{\beta^2\alpha^2}{\sqrt{n}},
\end{aligned}
$$

where (51) uses (50) and (52) is obtained by replacing $P^\top P$ with a matrix W and using that $\|P\|_F \le \beta \implies \|P^\top P\|_* = \|P\|_F^2 \le \beta^2$. (53) uses Claim 18, (54) follows because for any matrix A, $\|A\|_2 \le \|A\|_F$, and (55) follows from Jensen's inequality. Finally (56) follows by expanding the Frobenius norm and noting that the ξ_i are independent and $\mathbb{E}[\xi_i\xi_j] = 0$ for $i \ne j$. Combining the bounds for $(A), (B)$ and (C) we get the statement of the lemma. □

E.1 Small Weighted Square Loss on Test Samples from Train Manifolds

The first generalization bound is to show that on the m train manifolds, our learnt network achieves a small test error as measured by the weighted square loss. That is, the network has actually learnt to classify inputs from the m train manifolds. A simple uniform convergence argument coupled with Lemma 5 gives us that the expected weighted square loss over unseen samples from our train manifolds is small as well for large enough n.

Lemma 48. *At a local minimum $\hat{A}, \hat{B}$ we have, with probability $\ge 1-\delta$,*

$$
\frac{1}{m}\sum_{l=1}^{m}\mathop{\mathbb{E}}_{\mathbf{x}_l\sim\mathcal{D}(M_l)}\left[\|\mathbf{w_1}\odot(\mathbf{y_1}-\hat{\mathbf{y}}_\mathbf{1})\|_2^2\right] \le 2\varepsilon,
$$

for $n \ge \Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(m/\delta)}{\varepsilon^2}\right)$.

Proof. Let $\beta = s^{O(\log(1/\varepsilon)}$ be the bound we have on $\|\hat{B}\|_F$. Fix a train manifold with index l. Let $\hat{A}_l$ denote the l^{th} row of $\hat{A}$ and let $\|\hat{A}_l\|_2 \le a_l$. We know that $\sum_l a_l^2 \le O(m)$. Let $\mathbb{E}_n[\|\mathbf{w_1}\odot(\mathbf{y_1}-\hat{\mathbf{y}}_\mathbf{1})\|_2^2 \le \varepsilon_l$. From Lemma 26.5 of (Shalev-Shwartz & Ben-David, 2014) we have

that with probability $\geq 1-\delta/m$,

$$\mathop{\mathbb{E}}_{\mathbf{x}_l\sim\mathcal{D}(M_l)}\left[\|\mathbf{w_1}\odot(\mathbf{y_1}-\hat{\mathbf{y}}_\mathbf{1})\|_2^2\right]\leq\mathop{\mathbb{E}}_{n}\left[\|\mathbf{w_1}\odot(\mathbf{y_1}-\hat{\mathbf{y}}_\mathbf{1})\|_2^2\right]+2\,\mathbb{E}[\mathcal{R}_n(\mathcal{F})]+c\sqrt{\frac{2\log(m/\delta)}{n}},\quad(57)$$

In our case, $c=O(a_l^2\beta^2\|C\|_2^2)$ as $\|\mathbf{w_1}\odot(\mathbf{y_1}-\hat{\mathbf{y}}_\mathbf{1})\|_2^2\leq O(a_l^2\beta^2\|C\|_2^2)$ and $\mathcal{R}_n(\mathcal{F}_l)$ is the Rademacher complexity of the function class

$$\mathcal{F}_l=\left\{f(\mathbf{x_1})=\|\mathbf{w_1}\odot(\mathbf{y_1}-AB\sigma(C\mathbf{x_1}))\|_2^2\;\middle|\;\|A\|_F^2\leq O(m),\|B\|_F\leq\beta\right\}$$

Now $\|\mathbf{w_1}\odot(\mathbf{y_1}-AB\sigma(C\mathbf{x_1}))\|_2^2$ is of the form $\|P\mathbf{z}+\mathbf{b}\|_2^2$ for $P=\mathbf{w_1}\odot AB$, $\mathbf{b}=-\mathbf{w_1}\odot\mathbf{y_1}$ and $\mathbf{z}=\sigma(C\mathbf{x_1})$. Since, $\|P\|_F^2\leq O(a_l^2\beta^2)$, $\|\mathbf{b}\|_2=1/2$ and $\|\sigma(C\mathbf{x_1})\|_2\leq\|C\|_2$, from Lemma 47 we have that

$$\mathcal{R}_n(\mathcal{F}_l)\leq O\left(\frac{a_l^2\beta^2\|C\|_2^2}{\sqrt{n}}\right).$$

Therefore, we have that, with probability $\geq 1-\delta/m$,

$$\mathop{\mathbb{E}}_{\mathbf{x}_l\sim\mathcal{D}(M_l)}\left[\|\mathbf{w_1}\odot(\mathbf{y_1}-\hat{\mathbf{y}}_\mathbf{1})\|_2^2\right]\leq\mathop{\mathbb{E}}_{n}\left[\|\mathbf{w_1}\odot(\mathbf{y_1}-\hat{\mathbf{y}}_\mathbf{1})\|_2^2\right]+O\left(\frac{a_l^2\beta^2\|C\|_2^2}{\sqrt{n}}\right)+O(a_l^2\beta^2\|C\|_2^2)\sqrt{\frac{2\log(m/\delta)}{n}}.\quad(58)$$

Averaging over all m train manifolds and taking a union bound, we have with probability $\geq 1-\delta$,

$$\frac{1}{m}\sum_{l=1}^{m}\mathop{\mathbb{E}}_{\mathbf{x}_l\sim\mathcal{D}(M_l)}\left[\|\mathbf{w_1}\odot(\mathbf{y_1}-\hat{\mathbf{y}}_\mathbf{1})\|_2^2\right]\leq\varepsilon+O\left(\frac{\beta^2\|C\|_2^2}{\sqrt{n}}\right)+O(\beta^2\|C\|_2^2)\sqrt{\frac{2\log(m/\delta)}{n}}.\quad(59)$$

Note that we have used that $\sum_{l=1}^m a_l^2=O(m)$ here. (59) will imply the statement of the lemma for

$$n=\Theta\left(\frac{\beta^4\|C\|_2^4\log(m/\delta)}{\varepsilon^2}\right)=\Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(m/\delta)}{\varepsilon^2}\right),$$

with probability $\geq 1-\delta/2$. Here we used that for the D we have chosen $\|C\|_2$ is bounded by a constant with very high probability (Lemma 12). □

E.2 Generalization for Property (A)

We first show that the variance regularization term V_{reg} is an unbiased estimator for the intra-manifold variance of our representation.

Lemma 49 (Unbiased Variance Estimation)**.** *Consider the variance regularization term* (5)*. For the set of train manifolds $M_1,\ldots,M_m$, we have,*

$$\mathbb{E}[(5)]=\frac{1}{m}\sum_{l=1}^{m}V_{M_l}(B\sigma(C.)).$$

Proof. Let us focus on a single manifold l. Let $\mathbb{E}[\|B\mathbf{z}_l\|_2^2]=\mu_l$ and let $\|B\,\mathbb{E}[\mathbf{z}_l]\|_2^2=\kappa_l$.

$$\frac{1}{(n-1)}\,\mathbb{E}\left[\sum_{i=1}^{n}\|B\hat{\mathbf{z}}_{il}\|_2^2\right]=\frac{1}{n-1}\sum_{i=1}^{n}\left(\overbrace{\mathbb{E}\left[\|B\mathbf{z}_{il}\|_2^2\right]}^{(A)}+\overbrace{\mathbb{E}\left[\|B\mathop{\mathbb{E}}_{n}[\mathbf{z}_l]\|_2^2\right]}^{(B)}-\overbrace{2\,\mathbb{E}\left[\mathbf{z}_{il}^\top B^\top B\mathop{\mathbb{E}}_{n}[\mathbf{z}_l]\right]}^{(C)}\right).\quad(60)$$

Now $(A) = \mu_l$. We obtain the expressions for (B) and (C).

$$(B) = \frac{1}{n^2}\,\mathbb{E}\left[\left\|\sum_{j=1}^{n} B\mathbf{z}_{jl}\right\|_2^2\right] \tag{61}$$

$$= \frac{1}{n^2}\sum_{j=1}^{n}\mathbb{E}[\|B\mathbf{z}_{jl}\|_2^2] + \frac{1}{n^2}\sum_{j_1\neq j_2}\mathbb{E}\left[\mathbf{z}_{j_1 l}^\top B^\top B\mathbf{z}_{j_2 l}\right] \tag{62}$$

$$= \frac{\mu_l}{n} + \frac{n-1}{n}\,\mathbb{E}[\mathbf{z}_l]^\top B^\top B\,\mathbb{E}[\mathbf{z}_l] \tag{63}$$

$$= \frac{\mu_l}{n} + \frac{n-1}{n}\|B\,\mathbb{E}[\mathbf{z}_l]\|_2^2 = \frac{\mu_l}{n} + \frac{\kappa_l(n-1)}{n}. \tag{64}$$

Finally,

$$(C) = \frac{2}{n}\,\mathbb{E}\left[\sum_{j=1}^{n}\mathbf{z}_{il}^\top B^\top B\mathbf{z}_{jl}\right] \tag{65}$$

$$= \frac{2}{n}\,\mathbb{E}[\mathbf{z}_{il}^\top B^\top B\mathbf{z}_{il}] + \frac{2}{n}\sum_{j,j\neq i}\mathbb{E}[\mathbf{z}_{il}^\top B^\top B\mathbf{z}_{jl}] \tag{66}$$

$$= \frac{2\mu_l}{n} + \frac{2(n-1)\kappa_l}{n}. \tag{67}$$

Adding all together we get,

$$\frac{1}{(n-1)}\,\mathbb{E}\left[\sum_{i=1}^{n}\|B\hat{\mathbf{z}}_{il}\|_2^2\right] = \mu_l - \kappa_l. \tag{68}$$

It is easy to see via a similar calculation that,

$$V_{M_l}(B\sigma(C.)) = \mu_l - \kappa_l \tag{69}$$

$$\implies \frac{1}{(n-1)m}\,\mathbb{E}\left[\sum_{l=1}^{m}\sum_{i=1}^{n}\|B\hat{\mathbf{z}}_{il}\|_2^2\right] = \frac{1}{m}\sum_{l=1}^{m}V_{M_l}(B\sigma(C.)). \tag{70}$$

□

We next show that having a small variance regularization term over the train data implies that Property (A) of GSH holds with high probability.

Lemma 50 (Generalization of Property (A) to Unseen Points from Train Manifolds)**.** *Recall that at local minimum, we have found a $\hat{B}$ so that for $r(\mathbf{x}) = \hat{B}\sigma(C\mathbf{x})$,*

$$\frac{1}{m}\sum_{l=1}^{m}\frac{1}{n-1}\sum_{i=1}^{n}\|r(\mathbf{x_{il}}) - \mathop{\mathbb{E}}_{n}[r(\mathbf{x_l})]\|_2^2 = 0.$$

Then we have,

$$\mathbb{P}\left[\sum_{l=1}^{m}\frac{1}{m}V_{M_l}(r(.)) \leq 2\varepsilon\right] \geq 1-\delta, \tag{71}$$

where the probability is taken over the sampling of the n input examples from each of the m train manifolds.

Proof. From Lemma 49 we have that

$$\mathbb{E}\left[\frac{1}{n-1}\sum_{i=1}^{n}\|r(\mathbf{x_{il}}) - \mathop{\mathbb{E}}_{n}[r(\mathbf{x_l})]\|_2^2\right] = V_{M_l}(r(.)).$$

Given a vector $\mathbf{h} \in \mathbb{R}^D$, consider the family of functions defined as follows.

$$\mathcal{F}_{\mathbf{h}} = \left\{ f_{B,\mathbf{h}} : \mathbb{R}^d \to \mathbb{R} \;\middle|\; f_{B,\mathbf{h}}(x) = \|B\left(\sigma(C\mathbf{x}) - \mathbf{h}\right)\|_2^2, \|\mathbf{x}\|_2 \leq \alpha, \|B\|_F \leq \beta, \|\mathbf{h}\|_2 \leq \|C\|_2\alpha \right\}. \tag{72}$$

By Theorem 26.5 from (Shalev-Shwartz & Ben-David, 2014), we have that for each manifold M_l, for every $f_{B,\mathbf{h}} \in \mathcal{F}_{\mathbf{h}}$,

$$\mathop{\mathbb{E}}_{M_l}\left[f_{B,\mathbf{h}}(\mathbf{x})\right] \leq \mathop{\tilde{\mathbb{E}}}_{M_l}\left[f_{B,\mathbf{h}}(\mathbf{x})\right] + 2\mathcal{R}_n(\mathcal{F}_{\mathbf{h}}) + c\sqrt{\frac{2\log\left(m/\delta\right)}{n}}, \tag{73}$$

with probability $1 - \delta/m$. Here $|f_{B,\mathbf{h}}| \leq c$. For the function family we have considered taking $c = 4\beta^2\|C\|_2^2\alpha^2$ suffices. Note that for $\mathbf{h}(l) = \tilde{\mathbb{E}}_{M_l}[\sigma(C\mathbf{x})]$, $\mathbb{E}_{M_l}\left[f_{\tilde{B},\mathbf{h}(l)}(x)\right] = V_{M_l}(\tilde{B}\sigma(C.))]$ and $\sum_{l=1}^m \tilde{\mathbb{E}}_{M_l}\left[f_{B,\mathbf{h}(l)}(\mathbf{x})\right] = m(5) \leq m\varepsilon$. An upper bound on $\mathcal{R}_n(\mathcal{F}_{\mathbf{h}(l)})$ will lead us to an upper bound on $\sum_{l=1}^m V_{M_l}(\tilde{B}\sigma(C.))]$ as we shall see later. We proceed to bound $\mathcal{R}_n(\mathcal{F}_{\mathbf{h}(l)})$. For $\mathbf{x} \sim \mathcal{D}(M_l)$, let $\sigma(C\mathbf{x}) - \mathbf{h}(l) = \mathbf{z}$. From Lemma 47, we have

$$\mathcal{R}_n(\mathcal{F}_{\mathbf{h}(l)}) \leq \frac{3\beta^2\|C\|_2^2\alpha^2}{\sqrt{n}}.$$

Therefore, we get that

$$\begin{aligned}
\sum_{l=1}^m V_{M_l}(\hat{B}\sigma(C.)) &\leq \sum_{l=1}^m \mathop{\hat{\mathbb{E}}}_{M_l}\left[f_{B,h(l)}(x)\right] + 6m\beta^2\|C\|_2^2\alpha^2\sqrt{\frac{1}{n}} + 4m\beta^2\|C\|_2^2\alpha^2\sqrt{\frac{\log\left(m/\delta\right)}{n}} \\
&\leq m\varepsilon + 6m\beta^2\|C\|_2^2\alpha^2\sqrt{\frac{1}{n}} + 4m\beta^2\|C\|_2^2\alpha^2\sqrt{\frac{\log\left(m/\delta\right)}{n}} \\
&\leq m\varepsilon + 10m\beta^2\|C\|_2^2\alpha^2\sqrt{\frac{\log\left(m/\delta\right)}{n}}.
\end{aligned} \tag{74}$$

with probability $1 - \delta$. Note that we applied a union bound over the statement for each manifold bounding the probability of large deviation in any one of the manifolds by $m \cdot \delta/m = \delta$. By choosing $n \geq \Theta\left(\frac{\beta^4\alpha^4\|C\|_2^4\log(m/\delta)}{\varepsilon^2}\right)$ in (74), we get that with probability $1 - \delta$,

$$\frac{1}{m}\sum_{l=1}^m V_{M_l}(\hat{B}\sigma(C.)) \leq 2\varepsilon.$$

Since $\|\mathbf{x_l}\|_2 \leq 1$, we can choose $\alpha = 1$. Moreover from Lemma 12 we can assume $\|C\|_2$ is bounded by a constant with very high probability. In addition we have that at local minima $\|\hat{B}\|_F \leq s^{O(\log(1/\varepsilon))}$. Hence, we get the statement of the lemma for

$$n \geq \Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(m/\delta)}{\varepsilon^2}\right).$$

□

Lemma 51 (Generalization of the Property (A) to new Manifolds)**.** *Given $\hat{A}, \hat{B}$ which are at a local minima of either objective* (23) *or* (42)*, for a fresh sample $M_{m+1} \sim \mathcal{M}$, we have*

$$\mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}(M_{m+1})}\left[\left\|\hat{B}\mathbf{z}'\right\|_2^2\right] \leq O(\varepsilon), \tag{75}$$

with probability $\geq 9/10$ over the draw of M_{m+1} from $\mathcal{M}$ when

$$m \geq \Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(2/\delta)}{\varepsilon^2}\right).$$

Here $\mathbf{z}' = \sigma(C\mathbf{x}) - \mathbb{E}_{\mathbf{x}\sim\mathcal{D}(M_{m+1})}[\sigma(C\mathbf{x})]$.

Proof. Consider the following function class which maps a manifold to a non-negative real value:

$$\mathcal{F} = \left\{ f_B : M \to \mathbb{R}, f_B(M) = \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}(M)} \left[\|B\mathbf{z}'\|_2^2 \right] \;\middle|\; \|B\|_F \leq \beta, \|\mathbf{x}\|_2 \leq \alpha \right\}.$$

By the property of Rademacher complexity (Theorem 26.2 from (Shalev-Shwartz & Ben-David, 2014)), we have with probability $\geq 1-\delta$, for any $f \in \mathcal{F}$,

$$\mathop{\mathbb{E}}_{M_{m+1}\sim\mathcal{M}} [f(M_{m+1})] \leq \mathop{\mathbb{E}}_{m}[f(M_l)] + 2\,\mathbb{E}[\mathcal{R}_m(\mathcal{F})] + c\sqrt{\frac{\log(2/\delta)}{m}}, \tag{76}$$

with probability $1-\delta$. Here c is such that $|f(M)| \leq c$ for all M. Choosing $c = 4\beta^2\|C\|_2^2\alpha^2$ suffices. We need to bound $\mathcal{R}_m(\mathcal{F})$. We could appeal to Lemma 47 again but since now the function class $\mathcal{F}$ has functions of manifolds instead of vectors $\mathbf{x}$ we present the full argument here for clarity. The argument follows in a very similar vein to that of Lemma 47. Now we let $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_m)$ denote a vector of m i.i.d. Rademacher random variables.

$$\begin{aligned}
\mathcal{R}_m(\mathcal{F}) &= \frac{1}{m}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{f_B\in\mathcal{F}} \sum_{l=1}^m \xi_l f_B(M_l)\right] \\
&= \frac{1}{m}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{f_B\in\mathcal{F}} \sum_{l=1}^m \xi_l \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}(M_l)} \left[\|B\mathbf{z}'\|_2^2\right]\right] \\
&= \frac{1}{m}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{\|B\|_F\leq\beta} \sum_{l=1}^m \xi_l \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}(M_l)} \left[\left\langle B^\top B, \mathbf{z}'\mathbf{z}'^\top \right\rangle\right]\right] \\
&= \frac{1}{m}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{\|B\|_F\leq\beta} \left\langle B^\top B, \sum_{l=1}^m \xi_l \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}(M_l)} \left[\mathbf{z}'\mathbf{z}'^\top\right]\right\rangle\right] \\
&\leq \frac{1}{m}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\sup_{\|W\|_*\leq\beta^2} \left\langle W, \sum_{l=1}^m \xi_l \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}(M_l)} \left[\mathbf{z}'\mathbf{z}'^\top\right]\right\rangle\right] \\
&= \frac{\beta^2}{m}\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\left\|\sum_{l=1}^m \xi_l \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}(M_l)} \left[\mathbf{z}'\mathbf{z}'^\top\right]\right\|_2\right] \\
&\leq \frac{\beta^2}{m}\sqrt{\mathop{\mathbb{E}}_{\boldsymbol{\xi}}\left[\left\|\sum_{l=1}^m \xi_l \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}(M_l)} \left[\mathbf{z}'\mathbf{z}'^\top\right]\right\|_F^2\right]} \\
&= \frac{\beta^2}{m}\sqrt{\sum_{l=1}^m \left\|\mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}(M_l)} \left[\mathbf{z}'\mathbf{z}'^\top\right]\right\|_F^2} \\
&\leq \frac{\beta^2}{m}\sqrt{m\cdot\max\|\mathbf{z}'\|_2^4} \\
&\leq \frac{4\beta^2\alpha^2\|C\|_2^2}{\sqrt{m}}.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\mathop{\mathbb{E}}_{M_{m+1}\sim\mathcal{M}}\left[\mathop{\mathbb{E}}_{x\sim\mathcal{D}(M_{m+1})}\left[\|\hat{B}z'\|_2^2\right]\right] &\leq \varepsilon + \frac{8\beta^2\alpha^2\|C\|_2^2}{\sqrt{m}} + \frac{4\beta^2\alpha^2\|C\|_2^2\sqrt{\log(2/\delta)}}{\sqrt{m}} \\
&\leq \varepsilon + \frac{12\beta^2\alpha^2\|C\|_2^2\sqrt{\log(2/\delta)}}{\sqrt{m}} \\
&\leq 2\varepsilon,
\end{aligned}$$

for

$$m \geq \frac{144\beta^4\alpha^4\|C\|_2^4\log(2/\delta)}{\varepsilon^2} = \Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(2/\delta)}{\varepsilon^2}\right)$$

Finally, by Markov's inequality, we have that with probability $\geq 9/10$,

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}(M_{m+1})}\left[\|\hat{B}\mathbf{z}'\|_2^2\right] \leq 20\varepsilon.$$

□

E.3 Generalization for Property (B)

We first show that a population variant of Lemma 43 over n holds with high probability.

Lemma 52. *With probability $\geq 1-\delta$,*

$$\frac{1}{m(m-1)}\sum_{l=1}^{m}\sum_{j\neq l}\mathop{\mathbb{E}}_{\substack{\mathbf{x_l}\sim\mathcal{D}(M_l),\\ \mathbf{x_j}\sim\mathcal{D}(M_j)}}\|\hat{B}\sigma(C\mathbf{x_l})-\hat{B}\sigma(C\mathbf{x_j})\|_2^2 \geq \frac{m(1-O(\sqrt{\varepsilon}))}{O(\|A\|_F^2)} - f(\log(m)/\delta)/\sqrt{n}$$

Proof. Recall the definition of $H(l)$ from Lemma 43. Consider the sum $S=\frac{1}{m}\sum_{l=1}^{m}H(l)$. From this summation, consider the n terms corresponding to a pair of manifolds $l\neq j$. Denote the sum over these n terms by $S_{l,j}$. We can argue generalization for $S_{l,j}$ using uniform convergence theory. In particular, we have that with probability $1-\delta/(m(m-1))$,

$$S_{l,j} - \frac{1}{m(m-1)}\mathop{\mathbb{E}}_{\substack{\mathbf{x_l}\sim\mathcal{D}(M_l),\\ \mathbf{x_j}\sim\mathcal{D}(M_j)}}\|\hat{B}\sigma(C\mathbf{x_l})-\hat{B}\sigma(C\mathbf{x_j})\|_2^2 \tag{77}$$

$$\leq \quad 2\frac{\mathbb{E}[\mathcal{R}_n(\mathcal{F})]}{m(m-1)} + 4\beta^2\alpha^2\|C\|_2^2\frac{\sqrt{\log(2m(m-1)/\delta)}}{\sqrt{n}m(m-1)}, \tag{78}$$

where

$$\mathcal{F}=\{f_B : f_B(\mathbf{x_l},\mathbf{x_j}) = \|B\sigma(C\mathbf{x_l})-B\sigma(C\mathbf{x_j})\|_2^2, \|B\|_F\leq\beta\}$$

We repeat the above for all pairs $l\neq j$. The probability that for all pairs the expected loss will be close to the train loss is at least $1-\delta$ by the union bound. From (78) we have that with probability $\geq 1-\delta$,

$$S - \frac{1}{m(m-1)}\sum_{l=1}^{m}\sum_{j\neq l}\mathop{\mathbb{E}}_{\substack{\mathbf{x_l}\sim\mathcal{D}(M_l),\\ \mathbf{x_j}\sim\mathcal{D}(M_j)}}\|\hat{B}\sigma(C\mathbf{x_l})-\hat{B}\sigma(C\mathbf{x_j})\|_2^2 \tag{79}$$

$$\leq \quad 2\,\mathbb{E}[\mathcal{R}_n(\mathcal{F})] + 4\beta^2\alpha^2\|C\|_2^2\frac{\sqrt{\log(m(m-1)/\delta)}}{\sqrt{n}}$$

$$\implies \quad \frac{1}{m(m-1)}\sum_{l=1}^{m}\sum_{j\neq l}\mathop{\mathbb{E}}_{\substack{\mathbf{x_l}\sim\mathcal{D}(M_l),\\ \mathbf{x_j}\sim\mathcal{D}(M_j)}}\|\hat{B}\sigma(C\mathbf{x_l})-\hat{B}\sigma(C\mathbf{x_j})\|_2^2 \tag{80}$$

$$\geq \quad \frac{m(1-O(\sqrt{\varepsilon}))}{O(\|A\|_F^2)} - 2\mathcal{R}_n(\mathcal{F}) - 4\beta^2\alpha^2\|C\|_2^2\frac{\sqrt{\log(m(m-1)\delta)}}{\sqrt{n}}.$$

Now we bound $\mathcal{R}_n(\mathcal{F})$. Using Lemma 47, we get that

$$\mathcal{R}_n(\mathcal{F}) \leq \frac{4\beta^2\alpha^2\|C\|_2^2}{\sqrt{n}}. \tag{81}$$

Note that to employ Lemma 47 we think of $\sigma(C\mathbf{x_{il}}) - \sigma(C\mathbf{x_{ij}}) = \mathbf{x}$ as the input to the functions in $\mathcal{F}$. Therefore,

$$\frac{1}{m(m-1)}\sum_{l=1}^{m}\sum_{j\neq l}\mathop{\mathbb{E}}_{\substack{\mathbf{x_l}\sim\mathcal{D}(M_l),\\ \mathbf{x_j}\sim\mathcal{D}(M_j)}}\|\hat{B}\sigma(C\mathbf{x_l}) - \hat{B}\sigma(C\mathbf{x_j})\|_2^2 \geq \frac{m(1-O(\sqrt{\varepsilon}))}{O(\|A\|_F^2)} - \frac{8\beta^2\alpha^2\|C\|_2^2}{\sqrt{n}} \tag{82}$$

$$- 4\beta^2\alpha^2\|C\|_2^2\frac{\sqrt{\log(m(m-1)\delta)}}{\sqrt{n}} \tag{83}$$

$$\geq \frac{m(1-O(\sqrt{\varepsilon}))}{O(\|A\|_F^2)}, \tag{84}$$

for

$$n \geq \Theta\left(\frac{\beta^4\alpha^4\|C\|_2^4\log(m\delta)\|A\|_F^4}{m^2}\right) = \Theta\left(s^{O(\log(1/\varepsilon))}\log(m\delta)\right).$$

□

Next we show that for a randomly chosen permutation, with high probability, we have that the inter-manifold representation distance averaged over consecutive pairs according to the permutation is also large.

Lemma 53. *Suppose m is even and $m \geq \Theta(\log(2/\delta)s^{O(\log(1/\varepsilon))}/K^2)$. Let $d(l,j) = E_{\substack{\mathbf{x_l}\sim\mathcal{D}(M_l),\\ \mathbf{x_j}\sim\mathcal{D}(M_j)}}\|\hat{B}\sigma(C\mathbf{x_l}) - \hat{B}\sigma(C\mathbf{x_j})\|_2^2$. Suppose we have that $\frac{1}{m(m-1)}\sum_{l=1}^{m}\sum_{j\neq l}d(l,j) \geq K$. Then for a randomly chosen permutation $\rho : [m] \to [m]$, we have that with probability $\geq 1-\delta$,*

$$\frac{2}{m}\sum_{l=1}^{m/2} d(\rho(2l-1), \rho(2l)) \geq K/2.$$

Proof. First we have that

$$\mathop{\mathbb{E}}_{\rho}\left[\frac{2}{m}\sum_{l=1}^{m/2} d(\rho(2l-1),\rho(2l))\right] = \frac{1}{m(m-1)}\sum_{l=1}^{m}\sum_{j\neq l} d(l,j) \geq K. \tag{85}$$

This is because in expectation over random permutations, we see every pair l, j the same number of times. The normalization ensures that the overall sums match. Next, we show that concentration of the sum on the left hand side around its expected value. Using a trick from (Talagrand, 1995), we view the process of choosing a random permutation on $[m]$ as follows. We start with the identity permutation. Then we perform a sequence of $m-1$ transpositions as follows. We transpose (m, a_m), then $(m-1, a_{m-1})$ and so on till $(2, a_2)$ where each a_j is uniformly samples from $[j]$. This will give us a uniformly random permutation at the end and it is defined by $\{a_l\}_{l=2}^{m}$ which are independent. From here, our strategy will be to bound the amount by which our sum $S_m = \frac{2}{m}\sum_{l=1}^{m/2} d(\rho(2l-1),\rho(2l))$ changes when the value of some a_j is changed. Changing a_j changes at most 3 locations in the final permutation (wherever j, the old a_j, the new a_j end up). Therefore, at most 3 terms in S_m change. Noting that $\|B\sigma(C\mathbf{x_l}) - B\sigma(C\mathbf{x_j})\|_2 \leq 2\alpha\beta\|C\|_2$ we can deduce that by changing a single a_j, S_m changes by at most $\frac{6\alpha\beta\|C\|_2}{m}$. Now applying McDiarmid's inequality gives us

$$\mathop{\mathbb{P}}_{\rho}\left[\left|\frac{2}{m}\sum_{l=1}^{m/2} d(\rho(2l-1),\rho(2l)) - \mathop{\mathbb{E}}_{\rho}\left[\frac{2}{m}\sum_{l=1}^{m/2} d(\rho(2l-1),\rho(2l))\right]\right| \geq t\right] \leq 2\exp\left(-\frac{t^2 m}{36\alpha^2\beta^2\|C\|_2^2}\right). \tag{86}$$

Taking $t = \sqrt{\frac{\log(2/\delta)}{m}}6\alpha\beta\|C\|_2$, we get that with probability $1-\delta$,

$$\frac{2}{m}\sum_{l=1}^{m/2} d(\rho(2l-1),\rho(2l)) \geq K - t \geq K/2, \tag{87}$$

for $m = 144\log(2/\delta)\alpha^2\beta^2\|C\|_2^2/K^2 = \Theta(\log(2/\delta)s^{O(\log(1/\varepsilon))}/K^2)$. □

Finally we show property (B) of GSH for most new manifolds sampled from $\mathcal{M}$.

Lemma 54. *For $M_{m+1}, M_{m+2} \sim \mathcal{M}^2$, with probability $\geq 1-\delta$,*

$$E_{\substack{\mathbf{x_{m+1}}\sim\mathcal{D}(M_{m+1}),\\ \mathbf{x_{m+2}}\sim\mathcal{D}(M_{m+2})}} \|\hat{B}\sigma(C\mathbf{x_{m+1}}) - \hat{B}\sigma(C\mathbf{x_{m+2}})\|_2^2 \geq K/4,$$

for

$$m \geq \Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(2/\delta)}{K^2}\right).$$

Proof. Consider the following process. We sample $m/2$ pairs from $\mathcal{M}^2$, $\{M_l = (M_{l1}, M_{l2})\}_{l=1}^{m/2}$. Define $d(M_l) = \mathbb{E}_{\substack{\mathbf{x_1}\sim\mathcal{D}(M_{l1}),\\ \mathbf{x_2}\sim\mathcal{D}(M_{l2})}} \|\hat{B}\sigma(C\mathbf{x_1}) - \hat{B}\sigma(C\mathbf{x_2})\|_2^2$. It is easy to see that our original sampling process of getting $M_1, \ldots, M_m$ and choosing a random permutation to order these m manifolds in and pair consecutive ones is identical in distribution to the above described process. Hence, any probability statements for the former process hold also for the latter and vice versa. Let

$$\mathcal{F} = \{d_B : M \to \mathbb{R} \mid d_B(M) = \mathop{\mathbb{E}}_{\substack{\mathbf{x_1}\sim\mathcal{D}(M_1),\\ \mathbf{x_2}\sim\mathcal{D}(M_2)}} \|\hat{B}\sigma(C\mathbf{x_1}) - \hat{B}\sigma(C\mathbf{x_2})\|_2^2, \text{ where } M \in \mathcal{M}^2, \|B\|_F \leq \beta\}$$

We have that with probability $1-\delta$,

$$\sup_{d_B\in\mathcal{F}} \sum_{l=1}^{m/2} d_B(M_l) - \mathop{\mathbb{E}}_{M\sim\mathcal{M}^2}[d_B(M)] \leq 2\,\mathbb{E}[\mathcal{R}_{m/2}(\mathcal{F})] + 4\beta^2\alpha^2\|C\|_2^2\sqrt{\frac{2\log(2/\delta)}{m}}$$

$$\implies \mathop{\mathbb{E}}_{M\sim\mathcal{M}^2}[d_{\hat{B}}(M)] \geq K/2 - 2\,\mathbb{E}[\mathcal{R}_{m/2}(\mathcal{F})] - 4\beta^2\alpha^2\|C\|_2^2\sqrt{\frac{2\log(2/\delta)}{m}}.$$

Now it remains to bound $\mathcal{R}_{m/2}(\mathcal{F})$. Given $M \in \text{supp}(\mathcal{M})$ and $\mathbf{x_1}, \mathbf{x_2} \sim \mathcal{D}(M_1), \mathcal{D}(M_2)$ respectively, let $\mathbf{z} = \sigma(C\mathbf{x_1}) - \sigma(C\mathbf{x_2})$. Then we have $\mathbb{E}_{\mathbf{x_1},\mathbf{x_2}\sim\mathcal{D}(M_1),\mathcal{D}(M_2)}[\|\mathbf{z}\|_2^2] \leq 4\alpha^2\|C\|_2^2$.

$$\mathcal{R}_{m/2}(\mathcal{F}) = \frac{2}{m} \mathop{\mathbb{E}}_{\boldsymbol{\xi},\{M_l=(M_{l1},M_{l2})\}_{l=1}^{m/2}} \left[\sup_{B,\|B\|_F\leq\beta} \sum_{l=1}^{n} \xi_l \mathop{\mathbb{E}}_{\substack{\mathbf{x_1}\sim\mathcal{D}(M_{l1}),\\ \mathbf{x_2}\sim\mathcal{D}(M_{l2})}} \|B\mathbf{z}\|_2^2\right] \leq \frac{4\sqrt{2}\beta^2\alpha^2\|C\|_2^2}{\sqrt{m}}$$

using a line of calculations similar to those done in the proof of Lemma 51. Therefore, we have

$$\begin{aligned}\mathop{\mathbb{E}}_{M\sim\mathcal{M}^2}[d_{\hat{B}}(M)] &\geq K/2 - \frac{8\sqrt{2}\beta^2\alpha^2\|C\|_2^2}{\sqrt{m}} - 4\beta^2\alpha^2\|C\|_2^2\sqrt{\frac{2\log(2/\delta)}{m}} \\ &\geq K/2 - \frac{12\sqrt{2}\beta^2\alpha^2\|C\|_2^2\sqrt{\log(2/\delta)}}{\sqrt{m}} \geq K/4\end{aligned}$$

for

$$m \geq \frac{4608\beta^2\alpha^2\|C\|_2^2\log(2/\delta)}{K^2} = \Theta\left(\frac{s^{O(\log(1/\varepsilon))}\log(2/\delta)}{K^2}\right).$$

□

F Intra-Class Hashing Property Without Variance Regularization

Theorem 8 (Property (A) without Variance Regularization)**.** *Given our 3-layer neural network, and given n train samples each from m train manifolds, training the following objective results in a network that satisfies that $\hat{V}_{mn}(B\sigma(C\mathbf{x})) \to 0$ as $\lambda_1, \lambda_2 \to 0$*

$$\min_{A,B} \mathcal{L}_{A,B}(Y, \hat{Y}) + \lambda_1\|A\|_F^2 + \lambda_2\|B\|_F^2. \tag{88}$$

Proof. The main point to note is if $\lambda_1, \lambda_2 \to 0$ (that is very small) then the objective is dominated by $\mathcal{L}_{A,B}(Y, \hat{Y})$ which is minimized only if the prediction $\hat{Y}$ does not depend on $\boldsymbol{\theta}$ – because if it did then by replacing $\hat{Y}$ by $E_n[\hat{Y}]$ for each of the m manifolds, decreases the objective as shown in Lemma 36.

Now, we know that there is a ground truth model where $\hat{V}_{mn}(\hat{\mathbf{y}}) \leq \varepsilon$. Then it follows from Lemma 37 that $\hat{V}_{mn}(\mathbf{r}) \leq 2\sqrt{\varepsilon}$. To get close to this ground truth we select $\lambda_1 = \varepsilon/m$ and $\lambda_2 = \varepsilon/s^{O(\log(1/\varepsilon))}$. Hence by letting $\varepsilon \to 0$ we get the desired result.

□

G RECOVERING $\boldsymbol{\gamma}$ FROM THE REPRESENTATION

We have argued in Section 1 that in many cases where $\boldsymbol{\gamma}$ represents a set of semantic concepts such as the shape or texture of an image, it is of interest to recover exactly the latent vector $\boldsymbol{\gamma}$ and not just an isomorphism $f(\boldsymbol{\gamma})$. The next lemma shows that there is a linear transform that maps our learnt representation $r(\mathbf{x})$ to approximately $\boldsymbol{\gamma}$ associated with $\mathbf{x}$; however we can only show this with the unweighted square loss when the regularization weights λ_1, λ_2 is tiny and only for the train manifolds. Our experiments show that this reversibility holds even with our variant of the weighted square loss $\mathcal{L}_{A,B}(Y, \hat{Y})$.

Lemma 55 (Reversibility of the Learnt Representation)**.** *Consider the minimization* $\min_{A,B} \mathbb{E}\left[\|Y - AB\sigma(Cx))\|_F^2\right] + \lambda_1(\|A\|_F^2) + \lambda_2(\|B\|_F^2)$ *subject to* $v_\theta(r) = 0$*. As* $\lambda_1, \lambda_2 \to 0$ *and for infinite width* C *layer, there is a linear transform* R *so that* $R\hat{B}\sigma(C\mathbf{x_l}) = \boldsymbol{\gamma_l}$ *for any* $\mathbf{x_l}$ *from any of the* m *training manifolds.*

Proof. We will show that if $\boldsymbol{\gamma}$ is not expressible as a linear transform of $r(\mathbf{x})$ then creating additional outputs of the B layer that emit $\boldsymbol{\gamma}$ only improves the loss objective. First note that the width of the hidden layer never needs to be more than m (as otherwise we can replace A, B by their appropriate truncated-svd versions that are of at most width m since the rank of AB is at most m). So even if we add additional co-ordinates to $r(\mathbf{x})$ the width remains bounded. We have also assume that the variance at the representation layer is 0 for each manifold so the representation layer is a function only of the manifold for the train data.

Note that we have assumed $\lambda = 0$ and the width of the random ReLU layer goes to ∞. In this scenario, we know by Lemma 26 that for every manifold $M_l \in \mathcal{M}$, the representation computed by $\sigma(C\mathbf{x_l})$ when $\mathbf{x_l} \sim \mathcal{D}(M_l)$ is powerful enough to express $\boldsymbol{\gamma_l}$ exactly. We will show that if $\boldsymbol{\gamma_l}$ cannot be expressed as a linear combination of coordinates of $\mathbf{r} = \hat{B}\sigma(C\mathbf{x_l})$ over the training samples $\mathbf{x_l} \sim \mathcal{D}(M_l)$ then the loss $\mathbb{E}_{x_l \sim M_l}[\|Y - A\mathbf{r}\|_F^2]$ is not at a minimum and can be further reduced. Let A_l denote the l^{th} row of A. Let $\hat{A}_l$ be the regression minimization for the term $\min_{A_l} \|Y_l - A_l\mathbf{r}\|_2$ which will be the optimal trained value of A_l. Let us find the improvement to this term by appending $\boldsymbol{\gamma}^j$ the the j^{th} coordinate of $\boldsymbol{\gamma}$ which is the vector of γ_{lj} over the different manifolds l Let $\boldsymbol{\gamma}^j$ denote the vector of γ_{lj} over the different manifolds l. Note that in any linear regression problem the improvement obtained from a new coordinate of the input features can be quantified as follows: orthonormalize it with respect to the other coordinates and measure the square of the projection of the output vector along this new orthonormalized input coordinate. So when a new coordinate $\boldsymbol{\gamma}^j$ has been added to the input, the decrease in the square loss is $(\left\langle Y_l, \boldsymbol{\gamma}^{j'}/|\boldsymbol{\gamma}^{j'}|_2 \right\rangle)^2 = (Y_l^\top \boldsymbol{\gamma}^{j'})^2/|\boldsymbol{\gamma}^{j'}|_2^2$ where $\boldsymbol{\gamma}^{j'}$ is the component of $\boldsymbol{\gamma}^j$ that is orthogonal to $\mathbf{r}$; that is, $\boldsymbol{\gamma}^{j'} = \boldsymbol{\gamma}^j - d^\top \mathbf{r}$ so that $\langle \boldsymbol{\gamma}^{j'}, \mathbf{r} \rangle = 0$. Note that since $\boldsymbol{\gamma}^j$ have been normalized, the improvement is at least $(Y_l^\top \boldsymbol{\gamma}^{j'})^2 = \sum_{l=1}^m (\boldsymbol{\gamma}_l^{j'})^2$ (as Y_l is the indicator of the l^{th} coordinate). So the total improvement over all the manifolds from $\boldsymbol{\gamma}^j$ is at least $\sum_l (\boldsymbol{\gamma}_l^{j'})^2 = \|\boldsymbol{\gamma}^{j'}\|_2^2$. Now since the loss is at a local minimum it must be that there is no improvement possible which means $\|\boldsymbol{\gamma}^{j'}\|_2 = 0$ which means $\boldsymbol{\gamma}^j = \mathbf{a}^\top \mathbf{h}$ for some $\mathbf{a}$ and the same argument must be true for each coordinate of $\boldsymbol{\gamma}$ and so $\boldsymbol{\gamma} = R\mathbf{h}$ for some R.

□

Remark 2. *Although we assumed* $\lambda \to 0$ *and width of ReLU layer tends to* ∞*, note that if the width of* C *is bounded and large, then* $\boldsymbol{\gamma}$ *can only be approximately expressed in terms of* $\sigma(C\mathbf{x})$*. In that*

case that approximate version of $\boldsymbol{\gamma}$ must be linearly expressible in terms of $\mathbf{h}$.
Also even if λ is not 0, note that as long as $\sum_j \|\boldsymbol{\gamma}^{j'}\|_2^2 > \lambda$ the increase in regularization loss is more than offset by the decrease in square loss – to realize the improvement by $\sum_j \|\boldsymbol{\gamma}^{j'}\|_2^2$ the A matrix will add an edge of weight $\langle Y_i, \boldsymbol{\gamma}^{j'} \rangle$ between each $\boldsymbol{\gamma}^j$ and Y_i and the B matrix needs to add new nodes corresponding to b which has a bounded norm in terms of $\sigma(C\mathbf{x})$ which bounds the increase in Frobenius norm of B.

H Convergence of First or Second Order Methods to Local Optima

Here we point the reader to two results about some popular first-order optimization methods which have the property that they converge quickly to a local optimum for smooth optimization objectives. The first is the work of (Ge et al., 2015) which shows that for strictly-saddle objectives, a form of stochastic gradient descent provably converges to a local optimum. The second is the work of (Agarwal et al., 2017) who show that a second-order algorithm FastCubic converges to local optimum faster than gradient descent converges to any critical point for a set of smooth objectives which includes neural net training. There are more references within the above works studying similar properties of other variants as well.

I Additional Experimental Details

In this section, we support our theoretical results with an empirical study of the GSH property of DNNs on real and synthetic data. First, we detail our experimental setup and then discuss the experimental results.

I.1 Experimental Setup

We separate our experiments to two groups, based on the data generating process.

Natural Images. We train Myrtle-CNN (Page, 2018)—a five layer convolutional neural network—on MNIST (Deng, 2012) and CIFAR-10 (Krizhevsky et al., 2009) with ℓ_2 regularization without regularizing the bias terms. For CIFAR-10 the width parameter is $c = 128$ while for MNIST it is $c = 32$ and we remove the last two pooling layers. For both cases, we train via the SGD optimizer for 50 epochs with learning rate of 0.1 then drop the learning rate to 0.01 for another 100 epochs with batch size of 128. We use $\lambda = 0.1$ The resulting test accuracies are 99.4% for MNIST and 88.9% for CIFAR-10 while they also perfectly fit the train.

Synthetic Data. For the synthetic data we do the following data generating process, as to satisfy Assumption 1: first we randomly sample $\boldsymbol{\gamma}$ and $\boldsymbol{\theta}$ from the standard and $1/\sqrt{k}$ scaled Gaussians on $\mathbb{R}^k$ respectively for $k = 11$. Then, we sample two random matrices V, W from a scaled Gaussian $\mathcal{N}\left(0, \frac{1}{d} I\right)$ on $\mathbb{R}^{d \times k}$ and use an analytical function $\mathbf{p}$ such as the $\sin(\cdot)$ to generate $\mathbf{x} = \mathbf{p}(W\boldsymbol{\gamma} + U\boldsymbol{\theta})$. The analytic functions we tried are $e^{x/2}, \sin(x), \cos(x)$ and $\log((1+x^2)/2)$. Note that the last two are even functions, so precise recovery of $\boldsymbol{\gamma}$ is impossible as $f(x) = f(-x)$. To increase the complexity of the manifold we sum 4 functions of each type, so for example, the final sine data generating function is $\sum_{i=1^4} \sin(V_i\boldsymbol{\gamma} + W_i\boldsymbol{\theta})$. We also call a sum of all 4 functions the *Mixture distribution*. Now, in order to generate examples from the same manifold, we repeat the above process with $V, W, \boldsymbol{\gamma}$ fixed and vary (generate) $\boldsymbol{\theta}$. We then train a three layer Multi-Layer-Perceptron (MLP) with width 1000 for 200 epochs via SGD with learning rate 0.1, batch size of 32 with $\lambda = 0.01$-ℓ_2 regularization parameter and ℓ_2 loss for classification. The train/test accuracies after this procedure are 100%.

Meta learning and $\boldsymbol{\gamma}$ recovery. The main advantage of the synthetic data is that we are able to generate as many manifold (and samples) as we want. Therefore, we can check what is the ρ not only on manifolds we saw before, but also the behavior on the distribution. Moreover, we can generate enough manifolds to hope to fit a linear classifier on top of the representation. If the linear loss is

small the representation is approximately linearly isomorphic to $\boldsymbol{\gamma}$. This means that our representation successfully recovered the manifold geometry.

In a similar fashion described in the **Synthetic Data** data generation process, we generate 4 datasets: train, test, few-shot and few-shot-test. In the train (and test) there are 50 classes (manifolds) with $8k$ train examples and $2k$ test examples per class. As for the few-shot (and few-shot-test) we generate 10000 manifolds so their representations will serve as train set for the linear classifier that will try to recover $\boldsymbol{\gamma}$. In order to estimate the ρ on the distribution of unseen manifold, we sample 5 samples from each manifold to a total of 50000 few-shot (train) samples. We then use SGD with learning rate of 0.1 and batch size of 32 fit a linear model with loss $\|r(\mathbf{x}) - \boldsymbol{\gamma}\|_2^2$. Finally, we sample another 100 manifolds with one sample per manifold to estimate how well the linear function recovered $\boldsymbol{\gamma}$. In order to measure how close the representation is to an isomorphism, we use normalized distance as a metric (see table 1), normalizing $\|Wr(\mathbf{x}) - \boldsymbol{\gamma}\|_2^2$, where W is the learnt linear model, by the average distance between two $\boldsymbol{\gamma}$s: $\|\boldsymbol{\gamma}_i - \boldsymbol{\gamma}_j\|_2^2$. So a random Gaussian will produce normalized distance of 1 while perfect linear recovery will be normalized distance of 0.

I.2 Experimental Results

Our results for real data are in shown in Figure 4 and our results for synthetic data are summarized in Table 1. We observe high ρ values for both real (1.46 for CIFAR-10 and 3.36 for MNIST) and synthetic distributions ($\rho \geq 10.7$) both on train and on test. Furthermore, for synthetic data, we see that even out-of-distribution ρ is high. This implies that the classifier learnt is a GSH function on the population of manifolds, effectively inverting the data generating process. Further, we see that when the function is not even (and thus $\boldsymbol{\gamma}$ is not recoverable) as is the case for the Sine, Log, Exp and Mixture distributions, we are able to recover $\boldsymbol{\gamma}$ from the representation using a linear function. Specifically the normalized $\boldsymbol{\gamma}$ recovery distance is at ≈ 0.1 where a random γ would yield a distance of 1.

Table 1: Our results on synthetic data. We provide the ρ value for 5 different synthetic distributions, on train, test and transfer (i.e., unseen $\boldsymbol{\gamma}$s). We also note the normalized distance of $\|\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}\|$ where $\hat{\boldsymbol{\gamma}}$ is a linear classifier on top of the representation layer attempting to recover $\boldsymbol{\gamma}$. We are able to nicely recover the Mixture, Sine and Exp distributions, while recovering the $cos(x)$ and $\log((1+x^2)/2)$ proves more difficult. The reason is that we fail to recover $\boldsymbol{\gamma}$ for these functions is that they are even, so there is an ambiguity of whether the sign is positive or negative.

Data Generating Function	Test Accuracy	ρ-Train	ρ-Test	ρ-Transfer	Normalized $\boldsymbol{\gamma}$ Recovery Distance
Mixture	100%	19.13	19.09	20.46	0.11
Sine	100%	26.8	26.8	28.4	0.09
Cosine	100%	10.7	10.79	11.09	0.71
Log	100%	12.38	12.38	11.23	0.71
Exp	100%	25.2	25.2	28.77	0.08

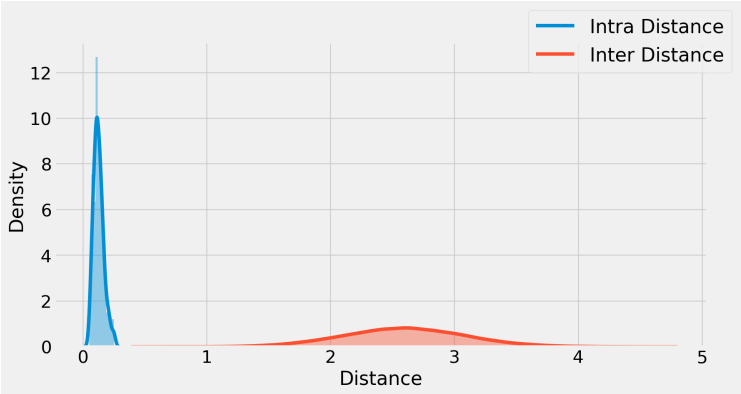

Figure 5: A comparison of intra vs inter class distances for *Unseen Manifold* for an MLP trained on Mixed synthetic data. Remarkably, even on unseen manifolds the GSH property holds, that is the representation is invariant to the "noisy feature" $\boldsymbol{\theta}$ while being sensitive to the semantically meaningful feature $\boldsymbol{\gamma}$.

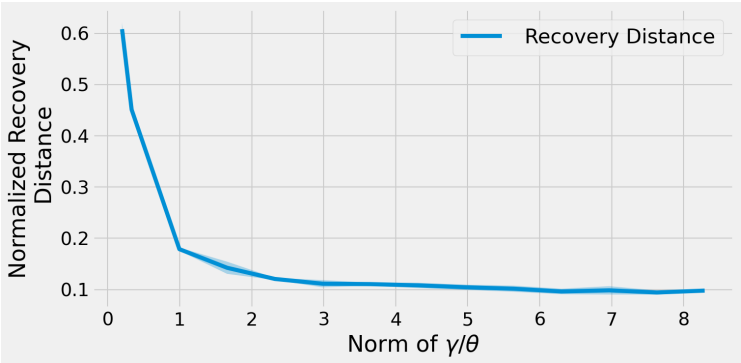

Figure 6: We show the ability to recover $\boldsymbol{\gamma}$ with linear regression over the representation with varying the norm of $\boldsymbol{\theta}/\boldsymbol{\gamma}$. We see that when the norm of $\boldsymbol{\gamma}$ is dominating, the learnt representation is almost a linear function of $\boldsymbol{\gamma}$. In contrast, when $\boldsymbol{\theta}$ has larger norm, the learnt representation becomes a non-linear function of $\boldsymbol{\gamma}$. (We know that $\boldsymbol{\gamma}$ and the representation are isomorphic as long as ρ is large enough.).