
Analyzing the Role of Permutation Invariance in Linear Mode Connectivity

Keyao Zhan*
Harvard University

Puheng Li*
Stanford University

Lei Wu†
Peking University

Abstract

It was empirically observed in [Entezari et al. \(2021\)](#) that when accounting for the permutation invariance of neural networks, there is likely no loss barrier along the linear interpolation between two SGD solutions – a phenomenon known as linear mode connectivity (LMC) modulo permutation. This phenomenon has sparked significant attention due to both its theoretical interest and practical relevance in applications such as model merging. In this paper, we provide a fine-grained analysis of this phenomenon for two-layer ReLU networks under a teacher-student setup. We show that as the student network width m increases, the LMC loss barrier modulo permutation exhibits a **double descent** behavior. Particularly, when m is sufficiently large, the barrier decreases to zero at a rate $O(m^{-1/2})$. Notably, this rate does not suffer from the curse of dimensionality and demonstrates how substantial permutation can reduce the LMC loss barrier. Moreover, we observe a sharp transition in the sparsity of GD/SGD solutions when increasing the learning rate and investigate how this sparsity preference affects the LMC loss barrier modulo permutation. Experiments on both synthetic and MNIST datasets corroborate our theoretical predictions and reveal a similar trend for more complex network architectures.

1 Introduction

Despite the remarkable successes of modern deep neural networks, a theoretical understanding of their underlying mechanisms remains elusive. One major challenge in deep learning theory is uncovering the structure of the high-dimensional loss landscape ([Wu et al., 2017](#); [Li et al., 2018b](#); [Zhang et al., 2021b](#); [Mei et al., 2018](#)), which is crucial for understanding the training dynamics. Unfortunately, the loss landscape exhibits significant complexity, high dimensionality, and degeneracy, characterized by numerous minima, symmetries, and saddle points ([Zhang et al., 2021a](#); [Draxler et al., 2018](#)). For instance, over-parameterized networks have the capacity to represent multiple functions that achieve similar performance on training data but differ significantly in parameter space ([Neyshabur et al., 2017](#); [Li et al., 2018a](#); [Liu et al., 2022](#)). Additionally, inherent scale and permutation invariances of neural networks allow a single function to be expressed through different parameter configurations within the same network. Despite these challenges, it is believed that the loss landscape encountered during practical training possess intricate yet benign property that facilitate the effectiveness of gradient-based optimization ([Ge et al., 2016](#); [Keskar et al., 2017](#)).

A particularly intriguing phenomenon uncovered in recent work is Mode Connectivity ([Freeman and Bruna, 2017](#); [Draxler et al., 2018](#); [Garipov et al., 2018](#)): Different optima found by independent gradient-based optimization runs turn out to be connected, i.e., there exists a path connecting them along which the loss or accuracy remains nearly constant. This is surprising because one would expect distinct optima of a non-convex function to lie in separate, isolated valleys—yet this separation does not occur in practice.

More recently, a stronger form of mode connectivity known as Linear Mode Connectivity (LMC) was proposed ([Frankle et al., 2020](#)): different optima can be connected by a linear path that does not pass through any loss barrier. LMC has since been used to explore various deep learning phenomena, such as generalization ([Juneja et al., 2022](#)). While LMC usually does

* The first two authors contributed equally.

† Corresponding author (leiwu@math.pku.edu.cn).

not emerge between two independently trained networks, it consistently appears in the following sense (Entezari et al., 2021; Ainsworth et al., 2022): For two independently trained global minima, there is a neuron permutation of one global minimum that makes it linearly connected to the other once the permutation is applied. However, despite these observations, this remains a conjecture awaiting a solid theoretical explanation.

Our contribution. In this paper, we provide a theoretical explanation of LMC modulo permutation for two-layer ReLU networks under a teacher-student setup (Lin et al., 2024). This setup allows us to quantify the influence of permutation invariance for LMC. Let m and M denote the numbers of neurons of the student and teacher networks, respectively. Our contributions are summarized as follows:

- First, we prove that applying permutations can **substantially** reduce the loss barrier of LMC, compared to direct linear interpolation without permutation. Specifically, the loss barrier modulo permutation diminishes to zero at a rate $O(m^{-1/2})$ when m is sufficiently large—crucially independent of the input dimension, thereby avoiding the curse of dimensionality. This offers a quantitative perspective on how permutations enhance LMC by lowering the loss barrier. In contrast, the upper bound $O(m^{-1/(2d+4)})$ in Entezari et al. (2021) (where d is the input dimension) suffers from the curse of dimensionality.
- Second, we provide a theoretical explanation for the “peak phenomenon” observed in Entezari et al. (2021), where the loss barrier (modulo permutation) initially increases and then decreases to zero as network width increases. We pinpoint the exact location of this peak in our setup. Moreover, we identify a **double descent** (Belkin et al., 2019) behavior in the LMC loss barrier (modulo permutation): it decreases as m increases up to $m = M$, then rises to a peak at $m = 2M$, before finally decreasing again.
- Third, we observe a sharp transition in the sparsity of GD/SGD solutions as the learning rate increases, and we further investigate how this preference for sparse solutions impacts the LMC modulo permutation.

1.1 Related Works

Mode connectivity. In the initial work Freeman and Bruna (2017), mode connectivity was proved for both linear networks and two-layer ReLU networks with ℓ_2 regularization. Garipov et al. (2018) and Fort and Jastrzebski (2019) empirically discovered

the piecewise-linear connecting paths. Nagarajan and Kolter (2019) first observed Linear Mode Connectivity (LMC), i.e., the near-constant-loss connecting path can be linear, on models trained on MNIST starting from the same random initialization. Later, Frankle et al. (2020) observed LMC in more difficult datasets, for networks that are jointly trained for a short period of time before going through independent training. Fort et al. (2020) explore the connection between LMC and the Neural Tangent Kernel dynamics. (Liang et al., 2018; Kuditipudi et al., 2019; Nguyen, 2019; Nguyen et al., 2018) provide some great insights into the geometry and connectivity of the loss landscape. More recent advancements, such as (Zhou et al., 2023; Ferbach et al., 2024), introduce new perspectives on LMC, including layerwise connectivity and optimal transport approaches.

Permutation invariance for LMC. As a conjecture proposed by Entezari et al. (2021), permutation invariance of linear mode connectivity has been constantly studied in recent works. Benzing et al. (2022) provided a simple algorithm for finding the optimal permutation and showed its connection with generalization. Ainsworth et al. (2022) showed that the global minima fall into a connected low-loss basin after permutation. Jordan et al. (2022) explored the limits of permutation that, in some regimes, permutation brings little improvement to linear mode connectivity.

2 Preliminaries

Notation. For $n \in \mathbb{N}$, let $[n] = \{1, 2, \dots, n\}$. For a compact set Ω , denote by $\text{Unif}(\Omega)$ the uniform distribution over Ω . Let $\{e_j\}_{j=1}^d$ be the canonical basis of \mathbb{R}^d . Let $\mathbb{S}^{d-1} = \{x \in \mathbb{R}^d : \|x\|_2 = 1\}$ and $\tau_{d-1} = \text{Unif}(\mathbb{S}^{d-1})$.

Denote by \mathcal{X} the input space, \mathcal{Y} the output space, and \mathcal{D} denote a data distribution over $\mathcal{X} \times \mathcal{Y}$. Let $f : \mathcal{X} \times \Theta \mapsto \mathcal{Y}$ be a neural network with Θ denoting the parameter space. For a given loss function $\ell : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$, the corresponding loss landscape is determined by

$$\mathcal{L}(\theta) = \mathbb{E}_{(x,y) \sim \mathcal{D}}[\ell(f(x;\theta), y)]. \quad (1)$$

In this paper, we focus on the over-realization regime where $\inf_{\theta \in \Theta} \mathcal{L}(\theta) = 0$. Then, the global minima manifold is given by

$$\mathcal{M} = \{\theta \in \Theta : \mathcal{L}(\theta) = 0\}. \quad (2)$$

Given that the parameter on the linear interpolation of two global minima θ_1 and θ_2 is not necessarily a global minima, we define the **linear barrier** between two minima (Frankle et al., 2020) as

$$B(\theta_1, \theta_2) = \sup_{\lambda \in [0,1]} [\mathcal{L}(\lambda\theta_1 + (1-\lambda)\theta_2)]$$

$$- [\lambda \mathcal{L}(\theta_1) + (1 - \lambda) \mathcal{L}(\theta_2)].$$

When empirically evaluated, we approximate this by evaluating $\lambda \in [0, 1]$ with a step size 0.1, calculating the barrier for $\lambda = 0.0, 0.1, 0.2, \dots, 1.0$ and taking the maximum value. We find in practice, this discretization is always sufficient, as the landscape along the linear interpolation is generally smooth and well-behaved. See Figure 8 for a few illustrations.

Model setup. We consider the two-layer ReLU network under the teacher-student setting, where the label is generated by a teacher network: $f^*(x) = \sum_{j=1}^M \sigma(w_j^* \cdot x)$ and the activation function is ReLU function defined by $\sigma(z) = \max(0, z)$. The number of teacher neurons is M , and the dimension of input is d , that is, $x \in \mathbb{R}^d$. We make the following assumption under this network architecture:

Assumption 1. Suppose $M \leq d$, $w_j^* = e_j$ for $j \in [M]$, and $x \sim \tau_{d-1}$.

By the rotational symmetry, this specific assumption is equivalent to only assuming $\{w_j^*\}_{j=1}^M$ to be orthonormal. In such a case, the loss objective function is

$$L(\theta) = \mathbb{E}_{x \sim \tau_{d-1}} \left[\left(\sum_{i=1}^m \sigma(w_i \cdot x) - \sum_{j=1}^M \sigma(x_j) \right)^2 \right], \quad (3)$$

where m denotes the number of neurons of the student network and $\theta = (w_1, w_2, \dots, w_m)^\top = (w_{i,j}) \in \mathbb{R}^{m \times d}$. Using this notation, each row of W represents a student neuron. We will utilize the following important conclusion of the global minima manifold of $L(\cdot)$ (Lin et al., 2024):

Lemma 2. Suppose that $m \geq M$. Let $S_0 = \{(0, \dots, 0) \in \mathbb{R}^d\}$, $S_j = \{\alpha e_j : \alpha > 0\}$ for $j \in [M]$, and $S = \cup_{j=0}^M S_j$. Then \mathcal{M} is compact and can be analytically characterized as follows

$$\mathcal{M} = \left\{ \theta = (w_1, \dots, w_m)^\top \in \mathbb{R}^{m \times d} : \forall i \in [m], w_i \in S \text{ and } \forall j \in [M], \sum_{i=1}^m w_{i,j} = 1 \right\}.$$

This characterization of the global minima manifold will play a critical role in our analysis. Note that $S_j \cap S_k = \emptyset$ for any $j \neq k \in \{0, 1, \dots, M\}$. Hence Lemma 2 implies the following facts about the global minima:

- There are at most $m + 1$ types of student neurons, represented by S_0, S_1, \dots, S_M , no matter how overparameterized the student network is. Moreover, for any $j \in [M]$, there exists at least one student neuron taking the type of S_j .

- For each neuron, there exists at most one coordinate to be nonzero and moreover, the coordinates from $M + 1$ to d must be zero.

3 Overlap Analysis

In this section, we will conduct an intuitive but effective analysis on how the barrier between two global minima changes with the number of student neurons m and teacher neurons M using the overlap between two minima.

According to Lemma 2, we denote the global minima θ as a weight matrix $W \in \mathbb{R}^{m \times d}$, and each row of this weight matrix is a neuron: $W = (w_1, w_2, \dots, w_m)^\top$, $w_i \in \mathbb{R}^d$. With previous characterization if the k -th component of a neuron is a non-zero element, then the row/ the neuron is said to belong to **Type k** , $k \in [M]$. We can also write $w_i \in S_j$ using the notation in Lemma 2.

We also denote $\alpha = (\alpha_1, \dots, \alpha_M)$ to be the **type vector** of the global minima W . As there is a constraint that for $\forall j \in [M]$ there exists at least one student neuron in type j , we let $\alpha_j + 1$ be the total number of neurons belonging to type j , or the non-zero elements in the k -th column of W , $k \in [M]$. Then $\sum_{i=1}^M \alpha_i = m - M$. To further analyze the overlap between two global minima, we make the following Uniform Distribution assumption of our solution obtained by GD/SGD on the loss landscape.

Assumption 3. (Uniform Distributed Solution) Each neuron is equally likely to be assigned to each type $S_j, j \in [M]$, that is, the type vector follows a multinomial distribution, $\alpha \sim \text{Multi}(m - M; \frac{1}{M}, \dots, \frac{1}{M})$. The actual number of neurons of type j is $\alpha_j + 1$. Moreover, the non-zero elements of neurons in any type follow a uniform distribution on the simplex.

In the absence of other prior knowledge, this uniform distribution assumption is natural, as all neurons are created equal, and the probability of neurons being allocated to each type should be the same. And we also assume that $\forall i \in [m], w_i \notin S_0$, that is, there is no sparsity for neurons. We will talk about the sparsity of global minima in Section 5, and we also validated this uniform distribution assumption through some simulation experiments in Appendix B.2. Here we set the parameter of multinomial distribution as $m - M$ to tackle the constraint that each type has at least one neuron.

The following matrix gives an example of a weight matrix (4) on the global minima manifold \mathcal{M} . We say that two global minima $W^{(1)}, W^{(2)}$ **match**, as long as their type vectors $\alpha^{(1)}, \alpha^{(2)}$ coincide. In other words, each row in these two weight matrices is the same type. It's easy to see that if $W^{(1)}, W^{(2)}$ match, then for

$\forall \lambda \in [0, 1], \lambda W^{(1)} + (1 - \lambda)W^{(2)} \in \mathcal{M}$, and the linear mode connectivity holds, which means the barrier between these two global minima is 0.

$$W = \begin{pmatrix} 1 & 2 & 3 & \cdots & M & M+1 & \cdots & d \\ 0 & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 0 & \frac{1}{2} & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \frac{1}{3} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2} & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{pmatrix} \begin{matrix} \text{Type 2} \\ \text{Type 3} \\ \\ \\ \text{Type M} \\ \\ \text{Type 1} \end{matrix} \quad (4)$$

However, as the type of neurons is uniformly randomly assigned, there is little chance that two global minima on the manifold have the same type of vectors. Therefore, we will analyze the degree of matching using the “**overlap**” of two global minima $W^{(1)}, W^{(2)}$.

Definition 4. Let $\alpha^{(1)}, \alpha^{(2)}$ be the type vectors of two solutions $W^{(1)}, W^{(2)}$, then the **overlap** between two type vectors $\alpha^{(1)}, \alpha^{(2)}$ (or two solutions) is defined as

$$C(\alpha^{(1)}, \alpha^{(2)}) := \sum_{j=1}^M \min(\alpha_j^{(1)}, \alpha_j^{(2)}) + M. \quad (5)$$

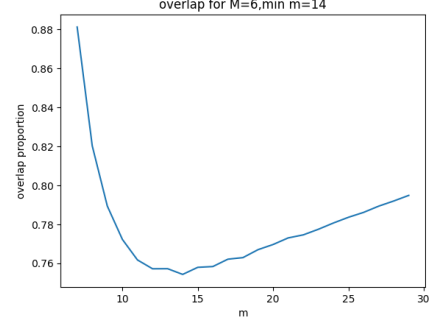
A neuron of a solution overlapping means it can be matched with a certain neuron in the other solution, so this pair incurs no barrier. Those not overlapped can’t be matched to any neuron in the other solution, incurring a high barrier. Therefore, the proportion of overlapping

$$P = C(\alpha^{(1)}, \alpha^{(2)})/m \quad (6)$$

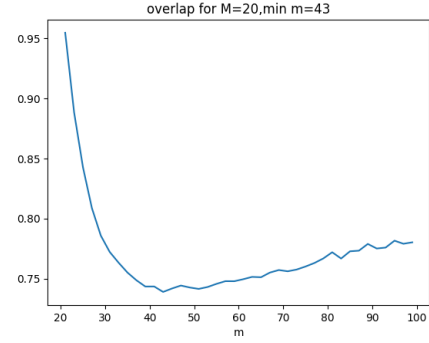
indicates a good property of permutation invariance. For instance, if $P = 1$, meaning two solutions can exactly match, then the barrier should be 0; on the other hand, if $P = 0.5$, say, then half of the neurons cannot be matched, incurring a high barrier.

Under the distributional Assumption 3, we can directly obtain how the overlap proportion of two solutions changes with the number of student neurons m :

Theorem 5. Assume two global minima $W^{(1)}, W^{(2)}$ follow the uniform distribution as in Assumption 3, and we fix the number of teacher neurons M , then their expected overlap proportion $T(m, M) := \mathbb{E}P = \frac{\mathbb{E}C(\alpha^{(1)}, \alpha^{(2)})}{m} \rightarrow 1$ when $m \rightarrow \infty$. Moreover, when $M \rightarrow \infty$, the limit value $\lim_{m, M \rightarrow \infty} T(m, M)$ is minimized with $\lim_{m, M \rightarrow \infty} \frac{m}{M} = 2$.



(a) $M = 6$



(b) $M = 20$

Figure 1: The overlap proportion curve for different M . The up panel shows the overlap proportion P for $M = 6$ and the down panel shows that for $M = 20$. Each data point is averaged over 10^4 simulations.

Theorem 5 means that in a limiting sense, the expected overlap proportion will be minimized when $m = 2M$. In Figure 1 we generated multinomial type vectors and calculated their overlap proportion as defined in (6). We can observe that the overlap decreases first and then increases as m increases, reaching its minimum value when m is approximately equal to $2M$; As m tends to infinity the proportion approaches 1. This result is consistent with the previous Theorem 5.

4 Barrier Calculation

When we consider the overlap of two global minima, we ignore that the non-zero elements also decay as m increases. If the non-overlap neurons have very small norms, the barrier will be very low even if the overlap proportion is away from 1. So we will employ a refined analysis on the barrier in this subsection.

Firstly, we need to figure out what kind of permutation we should adopt for two given solutions, in order to minimize the barrier as much as possible. The following algorithm tells us how to find such a permutation.

In the Algorithm 1, we try to find the best permutation

Algorithm 1: Find the best permutation for global minima $W^{(1)}, W^{(2)}$

Data: Two global minima $W^{(1)}$ and $W^{(2)}$

Result: Permuted $\widetilde{W}^{(2)}$

Sort non-zero elements of each column of $W^{(1)}, W^{(2)}$ in descending order;

Calculate the index set $I_j^{(1)}$ for W_1 ($j \in [M]$);

num[j]=1, for $j \in [M]$;

for $i = 1$ **to** m **do**

for $j = 1$ **to** M **do**

if $w_i^{(2)} \in S_j$ and $\text{num}[j] \leq |I_j^{(1)}|$ **then**

 index = $I_j^{(1)}[\text{num}[j]]$;

$\widetilde{W}^{(2)}[\text{index}, i] = w_i^{(2)}$;

 num[j] += 1;

end

end

end

Fill the unassigned neurons of $W^{(2)}$ sequentially into the empty rows of $\widetilde{W}^{(2)}$;

return $\widetilde{W}^{(2)}$;

of the neurons of $W^{(2)}$ so that the barrier of linear interpolation of $W^{(1)}$ and permuted $\widetilde{W}^{(2)}$ will be low. For each student neuron $w_i^{(2)} \in S_j$, we assign it to the corresponding type S_j of $W^{(1)}$ until all the neurons of j -th type in $W^{(1)}$ are already matched. Moreover, we sort the non-zero elements of $W^{(1)}, W^{(2)}$ first, so the neuron with a large norm will be matched first and the rest unassigned neurons of $W^{(2)}$ have a small norm.

In order to theoretically calculate the barrier after the best permutation, we use the kernel methods to analyze the loss function (3). Assume that $\pi = \rho = \tau_{d-1}$. By the rotational invariance, k_π can be written in a dot-product form:

$$k_\pi(x, x') = \int_{\mathbb{S}^{d-1}} \sigma(v^\top x) \sigma(v^\top x') d\tau_{d-1}(v) = \kappa(x^\top x'), \quad (7)$$

where $\kappa : [-1, 1] \rightarrow \mathbb{R}$. And for ReLU activation, we have (Cho and Saul, 2009; Wu and Long, 2022)

$$\kappa(t) = \frac{1}{2\pi d} \left((\pi - \arccos t)t + \sqrt{1 - t^2} \right). \quad (8)$$

From our loss function (3), we have the following derivation:

$$\begin{aligned} L(W) &= \mathbb{E}_{x \sim \tau_{d-1}} \left[\left(\sum_{i=1}^m \sigma(w_i^\top x) - \sum_{j=1}^M \sigma(e_j^\top x) \right)^2 \right] \\ &= \mathbb{E} \left[\left(\sum_{i=1}^m \sigma(w_i^\top x) \right)^2 + \left(\sum_{j=1}^M \sigma(e_j^\top x) \right)^2 \right] \end{aligned}$$

$$\begin{aligned} & - 2 \left(\sum_{j=1}^M \sigma(e_j^\top x) \right) \left(\sum_{i=1}^m \sigma(w_i^\top x) \right) \Bigg] \\ &= \sum_{i, i'} |w_i| |w_{i'}| \kappa(\tilde{w}_i^\top \tilde{w}_{i'}) + \sum_{j, j'} \kappa(e_j^\top e_{j'}) \\ & \quad - 2 \sum_{i, j} |w_i| \kappa(\tilde{w}_i^\top e_j), \end{aligned}$$

where \tilde{w}_i means the normalized vector with L_2 norm 1. With this characterization of the loss function, we have the following theoretical description of the barrier curve:

Theorem 6. For any two solutions $W^{(1)}, W^{(2)}$ following the Assumption 3, denote $\widetilde{W}^{(2)}$ as the permuted solution obtained by the Algorithm 1. The barrier is defined as

$$\begin{aligned} B(W^{(1)}, W^{(2)}) &:= \sup_{\lambda \in [0, 1]} \{L(\lambda W^{(1)} + (1 - \lambda)\widetilde{W}^{(2)}) \\ & \quad - \lambda L(W^{(1)}) - (1 - \lambda)L(W^{(2)})\}. \end{aligned} \quad (9)$$

Then we have $B(W^{(1)}, W^{(2)}) \rightarrow 0$ when $m \rightarrow \infty$, and the decay rate is $B(W^{(1)}, W^{(2)}) = O(m^{-1/2})$ ($m \rightarrow \infty$).

In order to characterize this rate of decrease in the barrier curve when $m \rightarrow \infty$, we need to assume the following approximation: for each type of neurons corresponding to every solution, the proportion of matching neurons is uniformly γ . The detailed meaning is as follows: Let $W^{(1)}, W^{(2)}$ be two solutions, then every neuron $w_i^{(1)}, w_i^{(2)} \in S_j$ ($j \in [M]$) as in Theorem 2. For simplicity, we just assume $W^{(2)}$ is already permuted by the Algorithm 1. Let $I_j^{(k)} = \{i : w_i^{(k)} \in S_j\}$, $j \in [M], k = 1, 2, I_j = I_j^{(1)} \cap I_j^{(2)}$. Then our assumption is

$$\sum_{i \in I_j} |w_i^{(k)}| = \gamma, \quad \forall j \in [M], k = 1, 2. \quad (10)$$

The detailed proof can be found in Appendix A.2.

We conducted simulation for the barrier of direct linear interpolation and the barrier after the best permutation. We also considered the minima obtained by SGD and the minima uniformly sampled from the manifold \mathcal{M} . Some detailed simulation settings can be found in Appendix B.1.

Figure 2 displays the changing trend of the barrier between two global minima found by SGD and those found by the uniform distribution on the manifold as m varies. We plot the barriers directly connecting the two global minima found by GD and those connecting after finding the optimal permutation using the

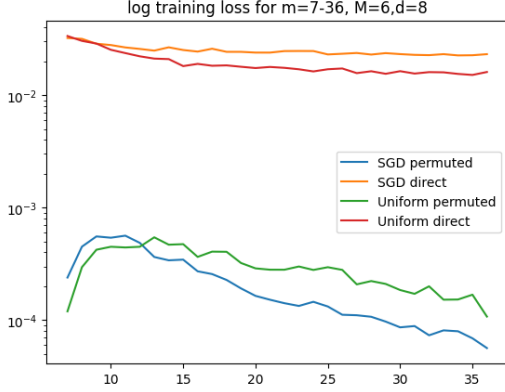


Figure 2: The log barrier curve for SGD solutions and uniformly sampled solutions. The number of teacher neurons $M = 6$, dimension is $d = 8$, and the number of student neurons m is varied from 7 to 36. Each data point is an average of 20 independent realizations.

aforementioned Algorithm 1. On the one hand, the barrier significantly decreases after the optimal permutation, demonstrating the correctness of permutation invariance. On the other hand, we can also observe that the barrier after permutation increases first and reaches its maximum when $m = 2M$; after that it decreases to 0 as m increases to infinity. This barrier curve obtained from simulation on the two-layer teacher-student ReLU network validates both the overlap Theorem 5 and the barrier Theorem 6. It’s also worth noting that when student neurons m is relatively large, the barrier of permuted SGD solutions is slightly smaller than the barrier of permuted uniformly sampled solutions. This is partly due to the sparsity of GD solutions when m is large and the learning rate is high. We will further discuss the sparsity of global minima in the following Section 5.

These phenomena are more evident in Figure 3, where we have normalized the barrier concerning the direct linear interpolation. We only use uniform samples in order to alleviate computational costs here, because the behavior of SGD solutions and uniformly sampled solutions are very similar, as can be seen in Figure 2. We can observe that the optimal permutation can reduce the barrier by 10^{-2} , and the phenomenon of the barrier reaching its maximum when $m = 2M$ becomes more evident as the number M of teacher neurons increases. Permutation also brings more benefits when M is larger. The barrier curve for more settings can be found in Appendix B.3.

Double descent. To further investigate the interplay between network size and loss barriers, we extend our analysis to the under-realization regime ($m < M$) and observe a clear “double descent” phenomenon

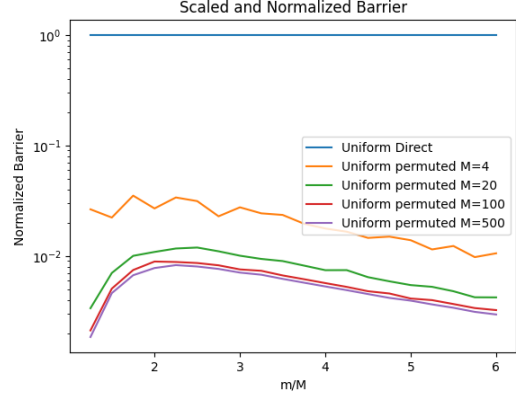


Figure 3: The normalized log barrier curve for uniformly sampled solutions. The barrier for direct linear interpolation in each setting with different M is normalized to 1, and we plot the relative barrier for permuted solutions with different numbers of teacher neurons $M = 4, 20, 100, 500$. x -axis is m/M and y -axis represents normalized barrier = $\text{Barrier}_{\text{Permuted}} / \text{Barrier}_{\text{Direct}}$. Each data point is an average of 50 independent realizations.

in Figure 4. Specifically, the first descent appears when m increases toward M . In this under-realization regime, each student solution can only align a subset of its neurons with those of the teacher, so two independently trained models tend to match different subsets of teacher neurons. This partial overlap leads to a non-trivial loss barrier between solutions that cannot be fully removed via permutation. As m approaches M , the extent of “unmatched” teacher neurons in each solution decreases, thereby lowering the barrier. The second descent occurs once m exceeds approximately $2M$, transitioning the student network into a regime where it has sufficient capacity to effectively match, and possibly surpass, the teacher neurons. The experimental results confirm both descents in the barrier size, demonstrating that the shift from under- to over-realization is key to understanding how network capacity influences solution alignment and, consequently, the loss landscape.

5 Sparsity of Global Minima

In previous discussions, we have been assuming under Assumption 3 that the solutions found by GD and SGD satisfy the property of uniform distribution. This assumption holds true when the learning rate of GD and SGD is relatively small. Some validation can be found in Appendix B.2.

However, in actual experiments we implemented, when the learning rate is large, we observe that GD/SGD tends to find sparser solutions, which means there are neurons whose elements are all zero.

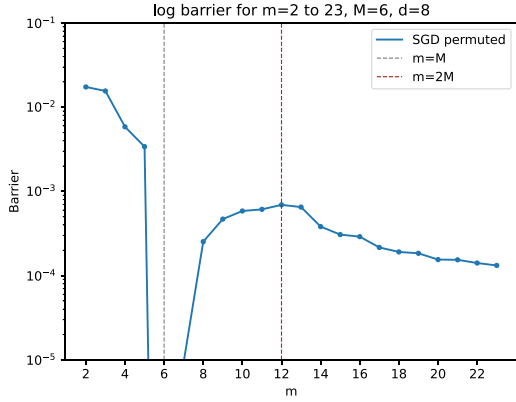


Figure 4: The double descent phenomenon for LMC modulo permutation. Barrier as a function of the number of student neurons m . The first descent appears as m approaches M (under-realization regime), and the second descent occurs as m exceeds $2M$, illustrating the “double descent” phenomenon. Note that when $m = M$, the student neurons can always match teachers and thus barrier is 0.

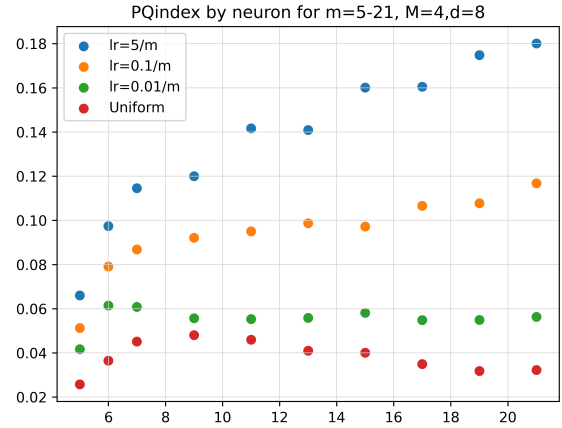


Figure 5: The PQI for the GD solution with different learning rate. x -axis is student neurons m and y -axis is PQI by neuron. **Uniform** means we uniformly generate samples from the manifold. Each data point is an average of 100 independent realizations.

To model the sparsity of a weight matrix W , we use the *PQ Index* (PQI) as a measure of sparsity. PQ Index describes the sparsity of a vector using the L_p norm, which is first proposed in [Diao et al. \(2023\)](#).

Definition 7. For any $0 < p < q$, the PQI of a non-zero vector $w \in \mathbb{R}^d$ is

$$\mathbf{I}_{p,q}(w) = 1 - d^{\frac{1}{q} - \frac{1}{p}} \frac{\|w\|_p}{\|w\|_q}, \quad (11)$$

where $\|w\|_p = \left(\sum_{i=1}^d |w_i|^p \right)^{1/p}$ is the ℓ_p -norm of w for any $p > 0$.

The larger the PQI is, the more sparse the vector will be. For example, for a unit vector $e_1 = (1, 0, 0, \dots, 0)$, which is very sparse, we have $\mathbf{I}_{p,q}(e_1) = 1 - d^{\frac{1}{q} - \frac{1}{p}}$; for a uniform vector $w = (1, 1, \dots, 1)$, we have $\mathbf{I}_{p,q}(w) = 0$. For our global minima, we have two different ways to evaluate its sparsity. On the one hand, we can directly flatten the matrix into a vector and then calculate the PQI of this vector. On the other hand, due to the properties of the solution, each neuron actually only has one non-zero element, so we can also take the norm by row first, and then calculate the PQI of this norm vector. In all of our simulations, we set $p = 0.5, q = 1$ in the PQI.

The sparsity of a solution obtained by GD/SGD is closely related to the learning rate, as indicated in the following experiments. In Figure 5, we plot the PQI of solutions obtained from GD with different learning rates. Note that we use learning rate that decays with the width m . We can clearly observe that with

a smaller learning rate the PQI of minima is smaller, and the uniformly sampled minima have the smallest PQI or sparsity. When the learning rate is large, the PQI also increases with the increase in network width m , while this phenomenon is not evident when the learning rate is low.

As in the previous theoretical analysis, we assume that GD/SGD solutions follow a uniform distribution as Assumption 3 and there is no zero row in W . Although a high learning rate or large m will encourage sparsity in the global minima, this sparsity is indeed beneficial for our desired permutation invariance, as those zero rows or zero neurons will incur no barrier when pairing, and the essential neurons causing overlap or barrier is smaller than total neuron number m . Therefore, in Figure 2 the barrier of permuted SGD solutions is lower than that of permuted uniformly sampled solutions, and we attribute its reason to the sparsity of the SGD solutions.

When we run gradient descent with different learning rates, we also observe a phase transition phenomenon of the sparsity of the solution. In Figure 6, we plotted the PQI of the solution obtained by GD without Nesterov with different learning rates. When the learning rate is below a certain threshold, it has little effect on sparsity, which manifests in the specific solution as having no zero rows and slowly increasing PQI. However, when the learning rate is increased beyond this threshold, for example, learning rate = 2 here, the resulting solution becomes sparse, zero rows appear in the solution, and the PQI also increases rapidly. This may be related to the loss landscape around global

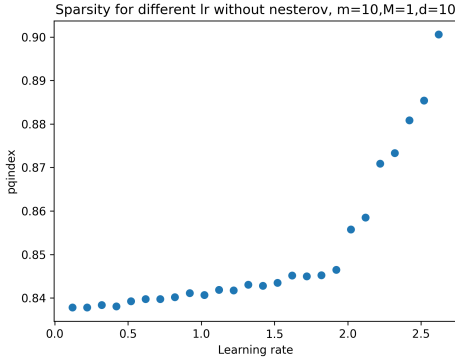


Figure 6: PQI of flattened W with different learning rates. We trained the network using GD without Nesterov, setting a grid for different learning rates from 0.1 to 2.0. y -axis is the PQ Index of flattened parameter matrix W .

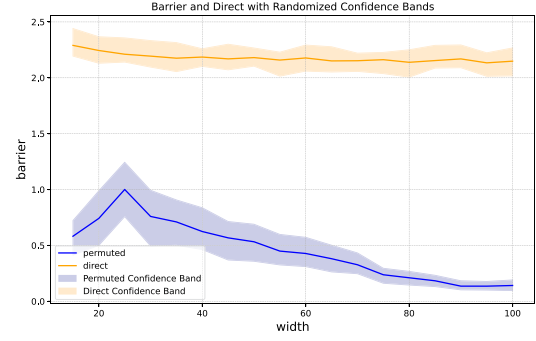


Figure 7: Barrier under different widths of the 4-layer MLP trained on MNIST with bands of top 90% and 10% percentile, for permuted and direct interpolation, respectively. Each result is an average of 10 independent realizations.

minima, and we look forward to future exploration providing a more detailed explanation of the threshold phenomenon in this setting.

6 Empirical Investigations

In this section, we empirically validate our theoretical findings beyond simulation data using two-layer teacher-student ReLU networks. For more complex architectures, such as multi-layer fully connected networks and CNNs, we apply the algorithms in [Benzing et al. \(2022\)](#); [Ainsworth et al. \(2022\)](#) to locate the best approximate permutation.

We first train 4-layer fully-connected neural networks for fitting the MNIST dataset, with a learning rate of 0.05. Figure 7 shows the LMC barrier (the negative log-likelihood) modulo permutation under different model widths from 15 to 100. It is clear that the barrier goes up and then goes down as the width increases, exhibiting a peak phenomenon. This is aligned with our theoretical analysis and previous simulation results.

Figure 8 further shows the interpolation plot comparing the loss on the line connecting original models and the line connecting permuted models. As we can see, the role of permutation invariance depends greatly on the model width. When the model width is $m = 25$, the effect of permutation is pretty limited, which is consistent with the peak value in Figure 7. For direct linear interpolation, the NLL is constantly large for all widths we have examined.

7 Conclusion and Discussion

In this paper, we analyzed the role of permutation invariance in a two-layer ReLU network under a teacher-student regime from a theoretical perspective. We

showed that as network width increases, the barrier of the linear connecting path between the permuted minima has a trend of first increasing and then decreasing to 0, with the maximum value at $m = 2M$. Further, we found that GD/SGD solutions have an increasing sparsity in learning rate with a phase transition pattern. Sparsity is beneficial to permutation invariance, hence this phenomenon serves as a complement to our theoretical analysis where we assume uniform distribution over global minima manifold. We empirically verified our results by conducting experiments on simulation data and MNIST datasets. The results explained why permutation invariance would appear significant or negligible under different conditions.

For future work, it remains an open problem why increasing learning rates yields sparser GD/SGD solutions. The role of permutation invariance with model depth is also a problem worth working on in the future. It is an intriguing question how the peak value of permuted barrier and the gap between permuted barrier and direct barrier change with width, depth, and network structure in various neural networks.

Acknowledgments

Lei Wu is supported by the National Key R&D Program of China (No. 2022YFA1008200) and National Natural Science Foundation of China (No. 12288101). We sincerely appreciate the constructive feedback from the anonymous reviewers.

References

Ainsworth, S. K., Hayase, J., and Srinivasa, S. (2022). Git re-basin: Merging models modulo permutation symmetries. *arXiv preprint arXiv:2209.04836*.

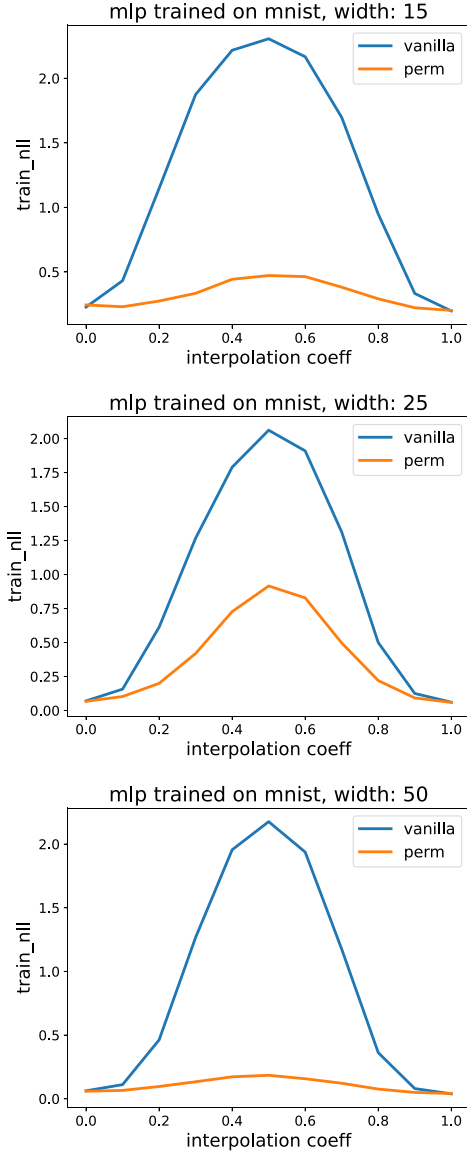


Figure 8: Interpolation NLL plot under different widths.

Belkin, M., Hsu, D., Ma, S., and Mandal, S. (2019). Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854.

Benzing, F., Schug, S., Meier, R., Von Oswald, J., Akram, Y., Zucchet, N., Aitchison, L., and Steger, A. (2022). Random initialisations performing above chance and how to find them. *arXiv preprint arXiv:2209.07509*.

Cho, Y. and Saul, L. (2009). Kernel methods for deep learning. *Advances in neural information processing systems*, 22.

Diao, E., Wang, G., Zhan, J., Yang, Y., Ding, J.,

and Tarokh, V. (2023). Pruning deep neural networks from a sparsity perspective. *arXiv preprint arXiv:2302.05601*.

Draxler, F., Veschgini, K., Salmhofer, M., and Hamprecht, F. (2018). Essentially no barriers in neural network energy landscape. In *International Conference on Machine Learning*, pages 1309–1318. PMLR.

Entezari, R., Sedghi, H., Saukh, O., and Neyshabur, B. (2021). The role of permutation invariance in linear mode connectivity of neural networks. *arXiv preprint arXiv:2110.06296*.

Ferbach, D., Goujaud, B., Gidel, G., and Dieuleveut, A. (2024). Proving linear mode connectivity of neural networks via optimal transport. In *International Conference on Artificial Intelligence and Statistics*, pages 3853–3861. PMLR.

Fort, S., Dziugaite, G. K., Paul, M., Kharaghani, S., Roy, D. M., and Ganguli, S. (2020). Deep learning versus kernel learning: an empirical study of loss landscape geometry and the time evolution of the neural tangent kernel. *Advances in Neural Information Processing Systems*, 33:5850–5861.

Fort, S. and Jastrzebski, S. (2019). Large scale structure of neural network loss landscapes. *Advances in Neural Information Processing Systems*, 32.

Frankle, J., Dziugaite, G. K., Roy, D., and Carbin, M. (2020). Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning*, pages 3259–3269. PMLR.

Freeman, C. D. and Bruna, J. (2017). Topology and geometry of half-rectified network optimization. In *5th International Conference on Learning Representations*.

Garipov, T., Izmailov, P., Podoprikin, D., Vetrov, D. P., and Wilson, A. G. (2018). Loss surfaces, mode connectivity, and fast ensembling of DNNs. *Advances in Neural Information Processing Systems*, 31.

Ge, R., Lee, J. D., and Ma, T. (2016). Matrix completion has no spurious local minimum. *Advances in Neural Information Processing Systems*, 29.

Jordan, K., Sedghi, H., Saukh, O., Entezari, R., and Neyshabur, B. (2022). Repair: Renormalizing permuted activations for interpolation repair. *arXiv preprint arXiv:2211.08403*.

Juneja, J., Bansal, R., Cho, K., Sedoc, J., and Saphra, N. (2022). Linear connectivity reveals generalization strategies. *arXiv preprint arXiv:2205.12411*.

Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., and Tang, P. T. P. (2017). On large-batch training for deep learning: Generalization gap and sharp

- minima. In *International Conference on Learning Representations*.
- Kuditipudi, R., Wang, X., Lee, H., Zhang, Y., Li, Z., Hu, W., Ge, R., and Arora, S. (2019). Explaining landscape connectivity of low-cost solutions for multilayer nets. *Advances in Neural Information Processing Systems*, 32.
- Li, D., Ding, T., and Sun, R. (2018a). On the benefit of width for neural networks: Disappearance of bad basins. *arXiv preprint arXiv:1812.11039*.
- Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. (2018b). Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31.
- Liang, S., Sun, R., Li, Y., and Srikant, R. (2018). Understanding the loss surface of neural networks for binary classification. In *International Conference on Machine Learning*, pages 2835–2843. PMLR.
- Lin, Z., Li, P., and Wu, L. (2024). Exploring neural network landscapes: Star-shaped and geodesic connectivity. *arXiv preprint arXiv:2404.06391*.
- Liu, C., Zhu, L., and Belkin, M. (2022). Loss landscapes and optimization in over-parameterized nonlinear systems and neural networks. *Applied and Computational Harmonic Analysis*, 59:85–116.
- Mei, S., Montanari, A., and Nguyen, P.-M. (2018). A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671.
- Nagarajan, V. and Kolter, J. Z. (2019). Uniform convergence may be unable to explain generalization in deep learning. *Advances in Neural Information Processing Systems*, 32.
- Neyshabur, B., Bhojanapalli, S., McAllester, D., and Srebro, N. (2017). Exploring generalization in deep learning. *Advances in neural information processing systems*, 30.
- Nguyen, Q. (2019). On connected sublevel sets in deep learning. In *International conference on machine learning*, pages 4790–4799. PMLR.
- Nguyen, Q., Mukkamala, M. C., and Hein, M. (2018). On the loss landscape of a class of deep neural networks with no bad local valleys. *arXiv preprint arXiv:1809.10749*.
- Wu, L. and Long, J. (2022). A spectral-based analysis of the separation between two-layer neural networks and linear methods. *J. Mach. Learn. Res.*, 23(1).
- Wu, L., Zhu, Z., et al. (2017). Towards understanding generalization of deep learning: Perspective of loss landscapes. *arXiv preprint arXiv:1706.10239*.
- Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2021a). Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115.
- Zhang, Y., Zhang, Z., Luo, T., and Xu, Z. J. (2021b). Embedding principle of loss landscape of deep neural networks. *Advances in Neural Information Processing Systems*, 34:14848–14859.
- Zhou, Z., Yang, Y., Yang, X., Yan, J., and Hu, W. (2023). Going beyond linear mode connectivity: The layerwise linear feature connectivity. *arXiv preprint arXiv:2307.08286*.

Checklist

- For all models and algorithms presented, check if you include:
 - A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
 - An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
 - (Optional) Anonymized source code, with the specification of all dependencies, including external libraries. [Not Applicable]
- For any theoretical claim, check if you include:
 - Statements of the full set of assumptions of all theoretical results. [Yes]
 - Complete proofs of all theoretical results. [Yes]
 - Clear explanations of any assumptions. [Yes]
- For all figures and tables that present empirical results, check if you include:
 - The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Not Applicable]
 - All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
 - A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
 - A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Not Applicable]
- If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:

- (a) Citations of the creator If your work uses existing assets. [Not Applicable]
 - (b) The license information of the assets, if applicable. [Not Applicable]
 - (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
 - (d) Information about consent from data providers/curators. [Not Applicable]
 - (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
- (a) The full text of instructions given to participants and screenshots. [Not Applicable]
 - (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
 - (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A Proofs

A.1 The Proof of Theorem 5

Based on the assumptions we have made, the marginal distribution of α_j is $\text{Bi}(m - M, 1/M)$, subject to the constraint that $\sum_{j=1}^M \alpha_j = m - M$. The α_j 's are identically distributed, though not independent. $\alpha^{(1)}, \alpha^{(2)}$ are independent. Then with the linearity of expectations, we have

$$T(m, M) := \frac{\sum_{j=1}^M \mathbb{E} \min(\alpha_j^{(1)}, \alpha_j^{(2)}) + M}{m} = \frac{M}{m} (\mathbb{E} \min(X, Y) + 1) \quad (12)$$

with X, Y i.i.d $\sim \text{Bi}(m - M, \frac{1}{M})$.

Equivalently we show that as a function of m , $T(m, M)$ from $m = M$ initially increases monotonically and then decreases monotonically. Here,

$$T(m, M) = \frac{M}{m} \left(\frac{m - M}{M} + 1 - \frac{1}{2} \mathbb{E} |X - Y| \right) = 1 - \frac{M}{2m} \mathbb{E} |X - Y|. \quad (13)$$

So we have the limit result. $(\mathbb{E} |X - Y|)^2 \leq \mathbb{E}(X - Y)^2 = 2 \text{Var } X = \frac{2(m - M)(M - 1)}{M^2}$. Hence $T(m, M) \geq 1 - \frac{\sqrt{2(m - M)(M - 1)}}{2m} \rightarrow 1$ when $m \rightarrow +\infty$.

With X, Y i.i.d $\sim \text{Bi}(m - M, \frac{1}{M})$, and $m, M \rightarrow \infty$, we have the following central limit theorem:

$$\frac{X - \frac{m - M}{M}}{\sqrt{(m - M) \frac{M - 1}{M^2}}} \rightarrow_d N(0, 1); \quad \frac{Y - \frac{m - M}{M}}{\sqrt{(m - M) \frac{M - 1}{M^2}}} \rightarrow_d N(0, 1) \quad (14)$$

For $\xi_1, \xi_2 \sim N(0, 1)$ i.i.d., $\mathbb{E} |\xi_1 - \xi_2| = \frac{2}{\sqrt{\pi}}$. Therefore

$$\lim_{m, M \rightarrow \infty} T(m, M) = 1 - \lim_{m, M \rightarrow \infty} \frac{M}{2m} \sqrt{\frac{(m - M)(M - 1)}{M^2}} \frac{2}{\sqrt{\pi}}. \quad (15)$$

Let $t = \lim_{m, M \rightarrow \infty} \frac{M}{m}$, then we have

$$\lim_{m, M \rightarrow \infty} T(m, M) = 1 - \sqrt{\frac{t(1 - t)}{\pi}}. \quad (16)$$

Then in this limiting sense, the overlap is minimized when $t = \frac{1}{2}$, which is $m = 2M$. \square

A.2 The Proof of Theorem 6

We first derive the loss function as

$$\begin{aligned} L(W) &= \mathbb{E}_{\mathbf{x} \sim \tau_{d-1}} \left[\left(\sum_{i=1}^m \sigma(w_i^\top x) - \sum_{j=1}^M \sigma(e_j^\top x) \right)^2 \right] \\ &= \mathbb{E} \left[\left(\sum_{i=1}^m \sigma(w_i^\top x) \right)^2 + \left(\sum_{j=1}^M \sigma(e_j^\top x) \right)^2 - 2 \left(\sum_{j=1}^M \sigma(e_j^\top x) \right) \left(\sum_{i=1}^m \sigma(w_i^\top x) \right) \right] \\ &= \sum_{i, i'} |w_i| |w_{i'}| \kappa(\tilde{w}_i^\top \tilde{w}_{i'}) + \sum_{j, j'} \kappa(e_j^\top e_{j'}) - 2 \sum_{i, j} |w_i| \kappa(\tilde{w}_i^\top e_j) \\ &=: L_1 + L_2 - 2L_3, \end{aligned}$$

where \tilde{w}_i means the normalized vector with L_2 norm 1. The simplest term is $\sum_{j, j'} \kappa(e_j^\top e_{j'})$. For $j = j'$, the inner product is 1; for $j \neq j'$, the inner product is 0. So we have $L_2 = \sum_{j, j'} \kappa(e_j^\top e_{j'}) = M\kappa(1) + (M^2 - M)\kappa(0)$.

We further assume that $I_c = \bigcup_{j=1}^M I_j$, $I_r = [m] \setminus I_c$. Now for an interpolation after best permutation, we permute the neurons of $W^{(2)}$ so that all neurons that can be matched are matched. We denote that $W := \lambda W^{(1)} + (1 - \lambda) \widetilde{W}^{(2)}$, where $\widetilde{W}^{(2)}$ means the solution after an appropriate permutation. Then we know that for $i \in I_j$, the neurons of interpolation also satisfy that $w_i \in S_j, \forall j \in [M]$. But for $i \in I_r$, the neuron w_i will have two non-zero elements, thus not belonging to any S_j .

Then we can write L_1 as

$$\begin{aligned}
 L_1 &= \sum_{i,i'}^m |w_i| |w_{i'}| \kappa(\tilde{w}_i^\top \tilde{w}_{i'}) \\
 &= \sum_{i,i' \in I_c} |w_i| |w_{i'}| \kappa(\tilde{w}_i^\top \tilde{w}_{i'}) + \sum_{i,i' \in I_r} |w_i| |w_{i'}| \kappa(\tilde{w}_i^\top \tilde{w}_{i'}) + 2 \sum_{i \in I_r, i' \in I_c} |w_i| |w_{i'}| \kappa(\tilde{w}_i^\top \tilde{w}_{i'}) \\
 &=: L_{11} + L_{12} + 2L_{13}. \\
 L_{11} &= \sum_{j=1}^M \sum_{i \in I_j} \left[\sum_{i' \in I_j} |w_i| |w_{i'}| \kappa(\tilde{w}_i^\top \tilde{w}_{i'}) + \sum_{i' \in I_{j'}, j' \neq j} |w_i| |w_{i'}| \kappa(\tilde{w}_i^\top \tilde{w}_{i'}) \right] \\
 &= \sum_{j=1}^M \sum_{i \in I_j} \left[\sum_{i' \in I_j} |w_i| |w_{i'}| \kappa(1) + \sum_{i' \in I_{j'}, j' \neq j} |w_i| |w_{i'}| \kappa(0) \right].
 \end{aligned}$$

For $w_i, i \in I_c$, $w_i \in S_j$ for some $j \in [M]$, so the norm of w_i is the value of its non-zero element. So we can derive that

$$\begin{aligned}
 L_{11} &= \sum_{j=1}^M \left[\sum_{i \in I_j} \sum_{i' \in I_j} |w_i| |w_{i'}| \kappa(1) + \sum_{i' \in I_{j'}, j' \neq j} |w_i| |w_{i'}| \kappa(0) \right] \\
 &= \sum_{j=1}^M [\gamma^2 \kappa(1) + (M-1) \gamma^2 \kappa(0)] = M \gamma^2 \kappa(1) + M(M-1) \gamma^2 \kappa(0).
 \end{aligned}$$

We can have the following estimation of the non-matching part I_r :

$$\sum_{i \in I_r} |w_i| = \sum_{i \in I_r} \sqrt{\sum_{j \in [M]} w_{ij}^2} \leq \sum_{i \in I_r} \sum_{j \in [M]} w_{ij} = M(1 - \gamma) \quad (17)$$

$$\geq \frac{\sqrt{2}}{2} \sum_{i \in I_r} \sum_{j \in [M]} w_{ij} = \frac{\sqrt{2}}{2} M(1 - \gamma). \quad (18)$$

because there are only two non-zero elements in each $w_i, i \in I_r$, so here the coefficient is $\frac{\sqrt{2}}{2}$. Then L_{12} and L_{13} can be estimated:

$$\begin{aligned}
 L_{12} &\leq \sum_{i,i' \in I_r} |w_i| |w_{i'}| \kappa(1) \leq M^2 (1 - \gamma)^2 \kappa(1), \\
 L_{13} &= \sum_{i \in I_r} \sum_{i' \in I_c} |w_i| |w_{i'}| \kappa(\tilde{w}_i^\top \tilde{w}_{i'}) \leq M \gamma (1 - \gamma) \kappa(1).
 \end{aligned}$$

Similarly, we can estimate the lower bound of L_3 :

$$\begin{aligned}
 L_3 &= \sum_{i,j} |w_i| \kappa(\tilde{w}_i^\top e_j) = \sum_{i \in I_c, j} |w_i| \kappa(\tilde{w}_i^\top e_j) + \sum_{i \in I_r, j} |w_i| \kappa(\tilde{w}_i^\top e_j) \\
 &= \sum_{j'=1}^M \sum_{i \in I_{j'}} |w_i| \left(\kappa(\tilde{w}_i^\top e_{j'}) + \sum_{j \neq j'} \kappa(\tilde{w}_i^\top e_j) \right) + \sum_{i \in I_r, j} |w_i| \kappa(\tilde{w}_i^\top e_j) \\
 &= M \gamma \kappa(1) + M(M-1) \gamma \kappa(0) + \sum_{i \in I_r, j} |w_i| \kappa(\tilde{w}_i^\top e_j)
 \end{aligned}$$

$$\geq M\gamma\kappa(1) + M(M-1)\gamma\kappa(0) + \frac{\sqrt{2}}{2}M^2(1-\gamma)\kappa(0).$$

In conclusion, we have an upper bound for the entire loss:

$$\begin{aligned} L &= L_1 + L_2 - 2L_3 \\ &\leq M\gamma^2\kappa(1) + M(M-1)\gamma^2\kappa(0) + M^2(1-\gamma)^2\kappa(1) + M\gamma(1-\gamma)\kappa(1) + M\kappa(1) + (M^2-M)\kappa(0) \\ &\quad - 2M\gamma\kappa(1) - 2M(M-1)\gamma\kappa(0) - \sqrt{2}M^2(1-\gamma)\kappa(0) \\ &= (1-\gamma)^2 [(M^2+M)\kappa(1) + (M^2-M)\kappa(0)] + (1-\gamma) [M\gamma\kappa(1) - \sqrt{2}M^2\kappa(0)] \\ &= O(1-\gamma) = O(m^{-1/2}). \end{aligned}$$

□

B Additional Experiment Results

B.1 Detailed Setting for Experiments

For two-layer ReLU network in the teacher-student regime, we conduct gradient descent without nesterov. We use random initialization, where each element is a Gaussian noise with standard error $1/md$. For the uniform distribution on the manifold, we utilize the following characterization:

Lemma 8. *Let (X_1, \dots, X_n) be a random point uniformly distributed on the simplex $\{(x_1, \dots, x_n) \mid \sum_{k=1}^n x_k = 1\}$. Then*

$$(X_1, \dots, X_n) \stackrel{d}{=} \frac{(Z_1, \dots, Z_n)}{Z_1 + \dots + Z_n},$$

where Z_1, \dots, Z_n are i.i.d $\text{Exp}(1)$ random variables. So, each X_i equals

$$\frac{Z_1}{Z_1 + \dots + Z_n} = \frac{Z_1}{n} / \frac{Z_1 + \dots + Z_n}{n}$$

in distribution. Also, $\frac{Z_1 + \dots + Z_n}{n} \rightarrow 1$ almost surely and hence in distribution (as $n \rightarrow \infty$), by the strong law of large numbers. Thus, for each i , the distribution of nX_i (not of X_i) goes to $\text{Exp}(1)$.

Therefore, we first generate the type vector $\alpha = (\alpha_1, \dots, \alpha_M)$ following multinomial distribution to determine the number of neurons in each type, and then generate exponentially distributed $\text{Exp}(1)$ for the non-zero element in each neuron. In the end, we normalize each type to one to make the solution on the manifold. It's also reasonable to use Dirichlet distribution or deterministic equal components as the data on the simplex.

For empirical investigation, we trained each global minimum with a 4-layer MLP with ReLU activations on MNIST dataset, with Kaiming-He initialization, SGD optimizer, batch size 100, and learning rate 0.05. Each minimum is trained with 10000 epochs. The widths for the network range from 15 to 100. For each reported value, we averaged the results for 10 independent realizations.

B.2 Validation of Uniform Distribution

In this subsection, we show some experiments validating our Assumption 3. In Figure 9a, our setting is $m = 6, M = 5, d = 5$. Therefore, each type vector must be a unit vector $e_j, j \in [5]$. If $\alpha \sim \text{Multi}(1; 1/5, 1/5, 1/5, 1/5, 1/5)$, then the probability of $\alpha = e_j$ is equally $1/5$. We run 500 independent experiments and obtain 500 GD solutions, and we count the number of $\alpha = e_j$. As can be seen in the figure, the number of 5 different type vectors are almost the same, which confirms the validity of the multinomial distribution assumption and also corroborates that each neuron indeed gets assigned to different classes with equal probability.

In Figure 9b, our setting is $m = 10, M = 1, d = 10$. Therefore, there is only one teacher neuron and all neurons is type 1. We plotted the weights of the first column of W , sorted from large to small (as other columns are all zeros). It can be seen that the distribution of these elements is consistent with sampling from an exponential distribution, so according to Lemma 8 earlier, it is also sampling from a uniform distribution on the simplex. However, it's worth noting that based on our extensive experiments, this distribution pattern is not stable enough with the changes in learning rate, m, M , and d . Therefore, it is also reasonable to model the distribution on this simplex in other ways.

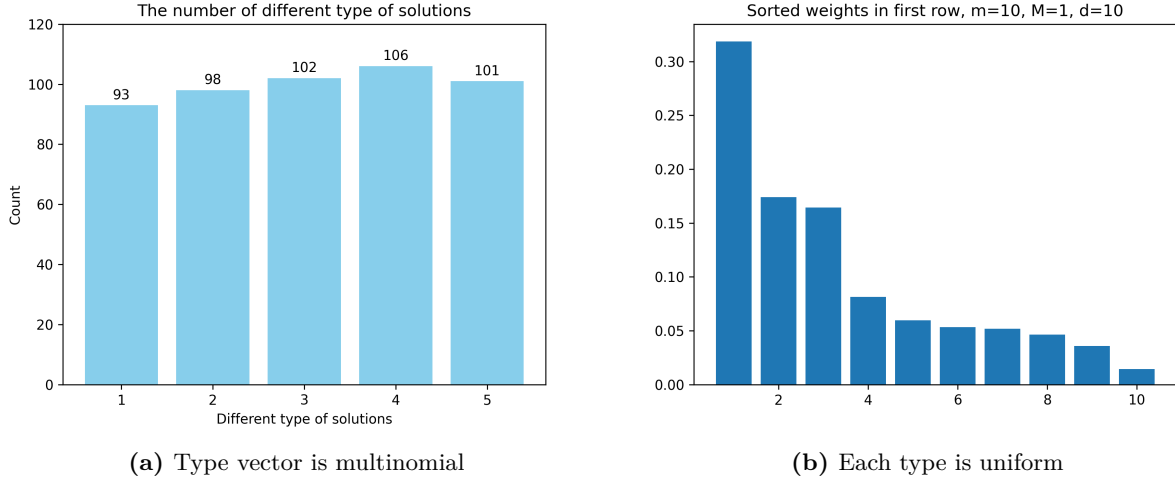


Figure 9: Evidence for Assumption 3

B.3 Barrier Curve in Different Settings

In this subsection we show some extra experiments in the simulation setting, giving comprehensive results for the behavior of the barrier of permuted minima and uniformly sampled solutions.

B.4 Peak Position

In our theoretical setting, we obtained the result that the peak of barrier curve occurred at $m = 2M$. Following the setting of empirical investigation in B.1, we studied the elements affecting the peak position empirically.

In Figure 14, we found that the peak position doesn't change according to depths, but it got bigger as our dataset became more complex. It aligns with our theoretical analysis in that the number of teacher neurons quantifies the difficulty of the task, which can be modeled by the complexity of the dataset.

B.5 Cifar10 Experiments

Figure 8 shows the barrier peak phenomenon of CNN trained on Cifar10. The result can also be verified by Figure 2 (left) provided in Entezari et al. (2021).

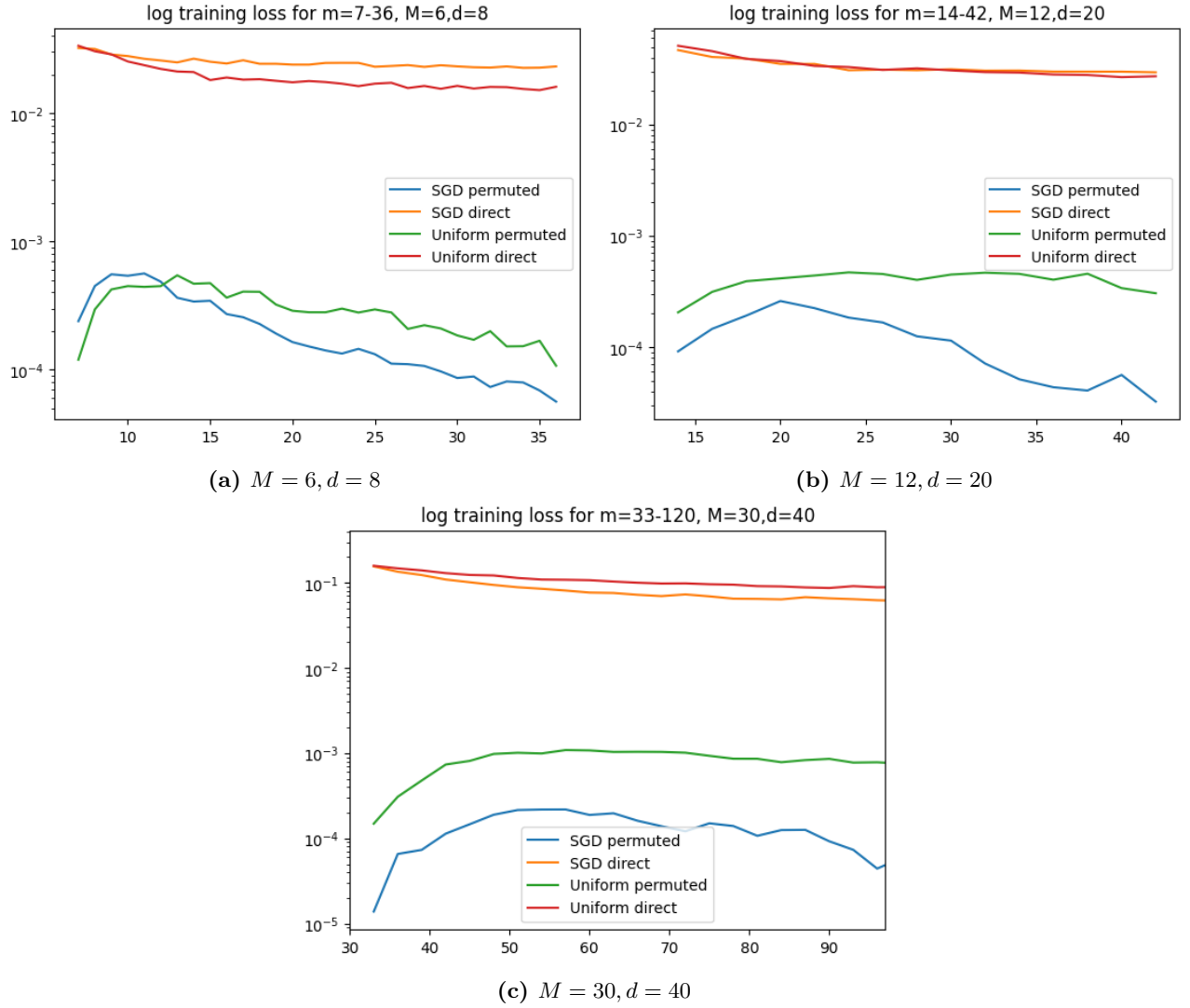


Figure 10: Barrier curve for different m, M, d . We increase M and d simultaneously. In each setting, we all can observe the trend of barrier first going up and then going down to 0. When M is larger, barrier of permuted SGD solutions is way more lower than that of uniformly sampled solutions. This is partly due to the sparsity of solution. Each data point here is an average of 20 independent realizations.

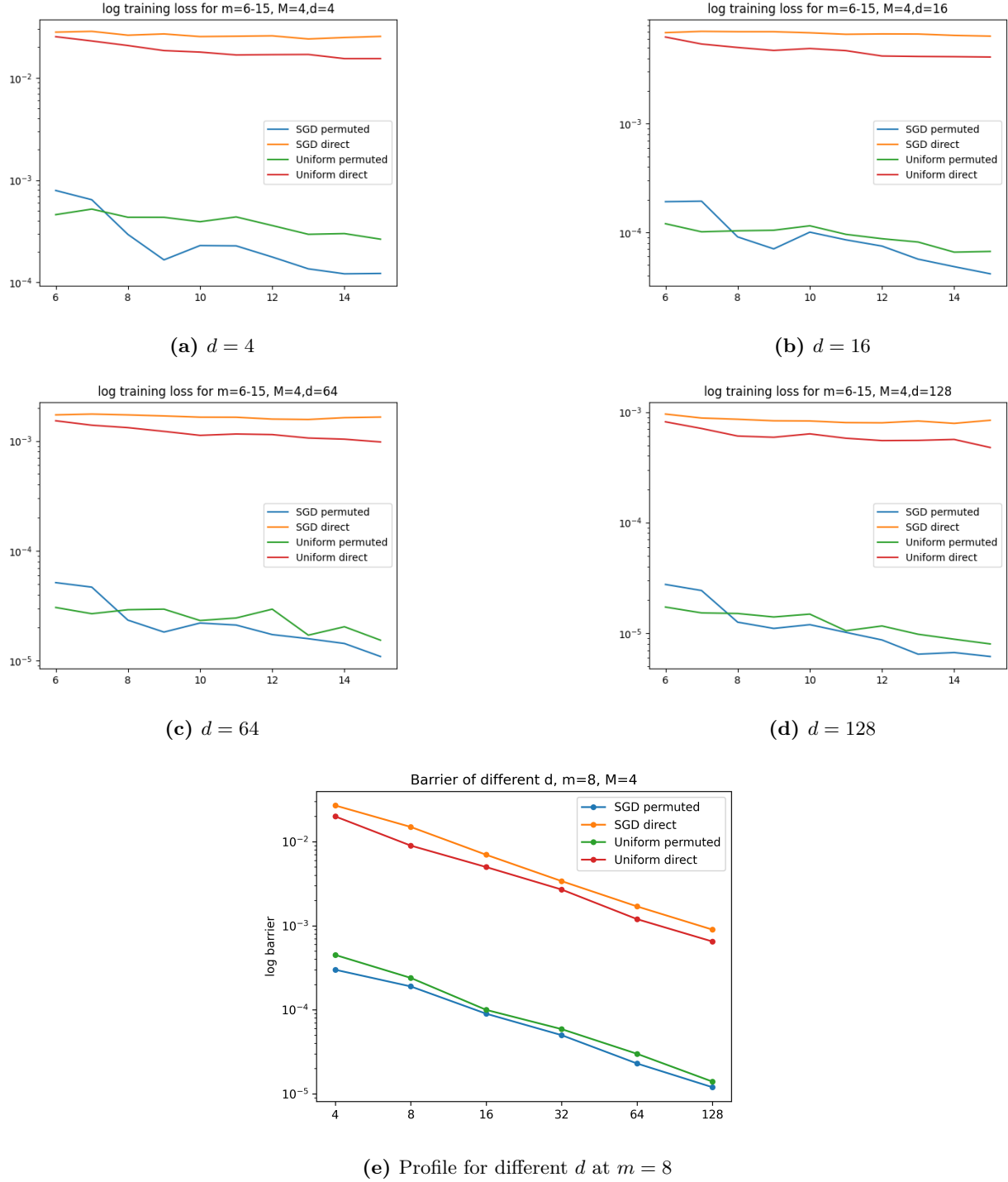


Figure 11: Barrier curve for different d with fixed $m \in [6, 15]$, $M = 4$. (a) - (d) give the curve for different m with increasing d , and (e) gives a profile at $m = 8$, $M = 4$ with exponentially increasing d from 2^2 to 2^7 . We can observe that the barrier is decreasing when d is going up, while the gap between the permuted barrier and the direct barrier remains almost the same.

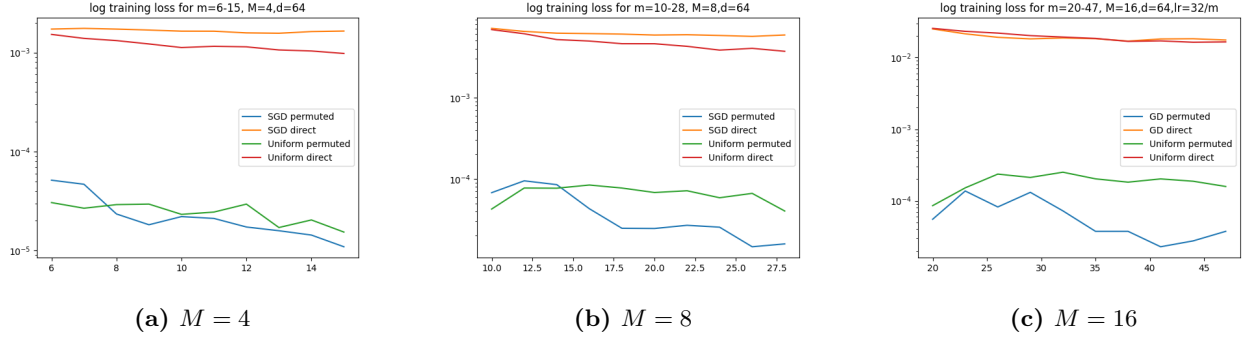


Figure 12: Barrier curve for different M with fixed $d = 64$. We can observe that the barrier is increasing when d is going up, while the gap between the permuted barrier and direct barrier remains also becomes larger.



Figure 13: Barrier of two permuted minima obtained by GD of different learning rates. The barrier first goes up and then goes down with increasing learning rate. Each data point is an average of 50 simulations.

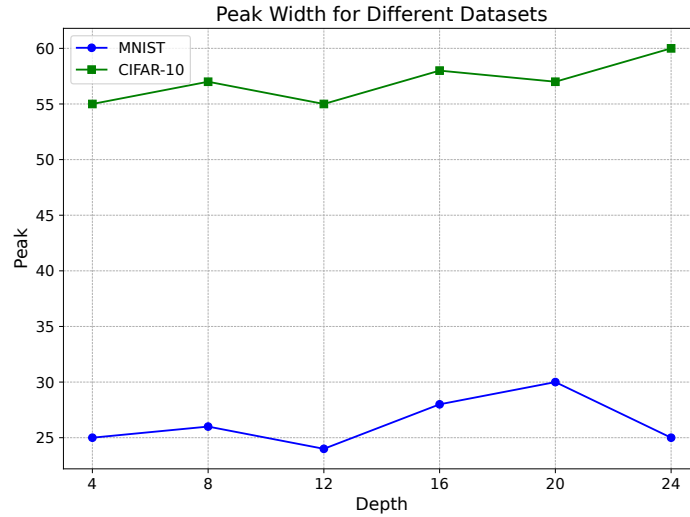


Figure 14: The peak widths for models trained on MNIST and Cifar10 with different depths.

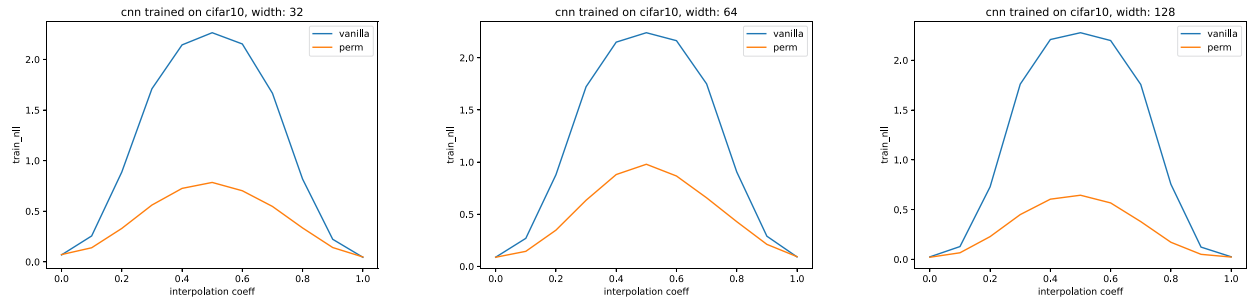


Figure 15: Interpolation NLL plot under different widths on CIFAR10. The network is CNN, with depth 16, optimizer Adam and learning rate 0.005. The barrier goes up and then goes down as the width increases, indicating the existence of a peak.