Supplement to: Embedding Principle of Loss Landscape of Deep Neural Networks

Yaoyu Zhang[1,2]*, **Zhongwang Zhang**[1] †, **Tao Luo**[1], **Zhi-Qin John Xu**[1]‡
[1] School of Mathematical Sciences, Institute of Natural Sciences, MOE-LSC and Qing Yuan Research Institute, Shanghai Jiao Tong University
[2] Shanghai Center for Brain Science and Brain-Inspired Technology
{zhyy.sjtu, 0123zzw666, luotao41, xuzhiqin}@sjtu.edu.cn.

A Appendix

Lemma (Lemma 1). *Given a L-layer ($L \geq 2$) fully-connected neural network with width $(m_0, \ldots, m_L)$, for any network parameters $\boldsymbol{\theta} = (\boldsymbol{W}^{[1]}, \boldsymbol{b}^{[1]}, \cdots, \boldsymbol{W}^{[L]}, \boldsymbol{b}^{[L]})$ and for any $l \in [L-1]$, $s \in [m_l]$, we have the expressions for $\boldsymbol{\theta}' := \mathcal{T}^{\alpha}_{l,s}(\boldsymbol{\theta})$ (see Fig. S2 for an illustration)*

(i) feature vectors in $\boldsymbol{F}_{\boldsymbol{\theta}'}$: $\boldsymbol{f}^{[l']}_{\boldsymbol{\theta}'} = \boldsymbol{f}^{[l']}_{\boldsymbol{\theta}}$, $l' \neq l$ and $\boldsymbol{f}^{[l]}_{\boldsymbol{\theta}'} = \left[(\boldsymbol{f}^{[l]}_{\boldsymbol{\theta}})^{\intercal}, (\boldsymbol{f}^{[l]}_{\boldsymbol{\theta}})_s\right]^{\intercal}$;

(ii) feature gradients in $\boldsymbol{G}_{\boldsymbol{\theta}'}$: $\boldsymbol{g}^{[l']}_{\boldsymbol{\theta}'} = \boldsymbol{g}^{[l']}_{\boldsymbol{\theta}}$, $l' \neq l$ and $\boldsymbol{g}^{[l]}_{\boldsymbol{\theta}'} = \left[(\boldsymbol{g}^{[l]}_{\boldsymbol{\theta}})^{\intercal}, (\boldsymbol{g}^{[l]}_{\boldsymbol{\theta}})_s\right]^{\intercal}$;

(iii) error vectors in $\boldsymbol{Z}_{\boldsymbol{\theta}'}$: $\boldsymbol{z}^{[l']}_{\boldsymbol{\theta}'} = \boldsymbol{z}^{[l']}_{\boldsymbol{\theta}}$, $l' \neq l$
and $\boldsymbol{z}^{[l]}_{\boldsymbol{\theta}'} = \left[(\boldsymbol{z}^{[l]}_{\boldsymbol{\theta}})^{\intercal}_{[1:s-1]}, (1-\alpha)(\boldsymbol{z}^{[l]}_{\boldsymbol{\theta}})_s, (\boldsymbol{z}^{[l]}_{\boldsymbol{\theta}})^{\intercal}_{[s+1:m_l]}, \alpha(\boldsymbol{z}^{[l]}_{\boldsymbol{\theta}})_s\right]^{\intercal}$.

Proof. (i) By the construction of $\boldsymbol{\theta}'$, it is clear that $\boldsymbol{f}^{[l']}_{\boldsymbol{\theta}'} = \boldsymbol{f}^{[l']}_{\boldsymbol{\theta}}$ for any $l' \in [l-1]$. Then

$$\boldsymbol{f}^{[l]}_{\boldsymbol{\theta}'} = \sigma \circ \left(\begin{bmatrix} \boldsymbol{W}^{[l]} \\ \boldsymbol{W}^{[l]}_{s,[1:m_{l-1}]} \end{bmatrix} \boldsymbol{f}^{[l-1]}_{\boldsymbol{\theta}} + \begin{bmatrix} \boldsymbol{b}^{[l]} \\ \boldsymbol{b}^{[l]}_{s} \end{bmatrix} \right) = \begin{bmatrix} \boldsymbol{f}^{[l]}_{\boldsymbol{\theta}} \\ (\boldsymbol{f}^{[l]}_{\boldsymbol{\theta}})_s \end{bmatrix}. \tag{1}$$

Note that

$$\alpha \left[\boldsymbol{0}_{m_{l+1} \times (s-1)}, -\boldsymbol{W}^{[l+1]}_{[1:m_{l+1}],s}, \boldsymbol{0}_{m_{l+1} \times (m_l - s)}, \boldsymbol{W}^{[l+1]}_{[1:m_{l+1}],s} \right] \begin{bmatrix} \boldsymbol{f}^{[l]}_{\boldsymbol{\theta}} \\ (\boldsymbol{f}^{[l]}_{\boldsymbol{\theta}})_s \end{bmatrix} = \boldsymbol{0}_{m_{l+1} \times 1}.$$

Thus

$$\boldsymbol{f}^{[l+1]}_{\boldsymbol{\theta}'} = \sigma \circ \left(\left[\boldsymbol{W}^{[m_{l+1}]}, \boldsymbol{0}_{m_{l+1} \times 1} \right] \begin{bmatrix} \boldsymbol{f}^{[l]}_{\boldsymbol{\theta}} \\ (\boldsymbol{f}^{[l]}_{\boldsymbol{\theta}})_s \end{bmatrix} + \boldsymbol{0}_{m_{l+1} \times 1} + \boldsymbol{b}^{[l+1]} \right) = \boldsymbol{f}^{[l+1]}_{\boldsymbol{\theta}}. \tag{2}$$

Next, by the construction of $\boldsymbol{\theta}'$ again, it is clear that $\boldsymbol{f}^{[l']}_{\boldsymbol{\theta}'} = \boldsymbol{f}^{[l']}_{\boldsymbol{\theta}}$ for any $l' \in [l+1:L]$.

(ii) The results for feature gradients $\boldsymbol{g}^{[l']}_{\boldsymbol{\theta}'}$, $l' \in [L]$ can be calculated in a similar way.

(iii) By the backpropagation and the above facts in (i), we have $\boldsymbol{z}^{[L]}_{\boldsymbol{\theta}'} = \nabla \ell(\boldsymbol{f}^{[L]}_{\boldsymbol{\theta}'}, \boldsymbol{y}) = \nabla \ell(\boldsymbol{f}^{[L]}_{\boldsymbol{\theta}}, \boldsymbol{y}) = \boldsymbol{z}^{[L]}_{\boldsymbol{\theta}}$.

*Corresponding author: zhyy.sjtu@sjtu.edu.cn.

†Part of this work is done when ZZ was an undergraduate student of Zhiyuan Honors Program at Shanghai Jiao Tong University.

‡Corresponding author: xuzhiqin@sjtu.edu.cn.

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

Recalling the recurrence relation for $l' \in [l+1 : L-1]$, then we recursively obtain the following equality for l' from $L-1$ down to $l+1$:

$$\boldsymbol{z}_{\boldsymbol{\theta}'}^{[l']} = (\boldsymbol{W}^{[l'+1]})^\intercal \boldsymbol{z}_{\boldsymbol{\theta}'}^{[l'+1]} \circ \boldsymbol{g}_{\boldsymbol{\theta}'}^{[l'+1]} = (\boldsymbol{W}^{[l'+1]})^\intercal \boldsymbol{z}_{\boldsymbol{\theta}}^{[l'+1]} \circ \boldsymbol{g}_{\boldsymbol{\theta}}^{[l'+1]} = \boldsymbol{z}_{\boldsymbol{\theta}}^{[l']}. \tag{3}$$

Next,

$$\begin{aligned}
\boldsymbol{z}_{\boldsymbol{\theta}'}^{[l]} &= \left(\left[\boldsymbol{W}^{[m_{l+1}]}, \boldsymbol{0}_{m_{l+1}\times 1} \right] + \alpha \left[\boldsymbol{0}_{m_{l+1}\times(s-1)}, -\boldsymbol{W}^{[l+1]}_{[1:m_{l+1}],s}, \boldsymbol{0}_{m_{l+1}\times(m_l-s)}, \boldsymbol{W}^{[l+1]}_{[1:m_{l+1}],s} \right] \right)^\intercal \boldsymbol{z}_{\boldsymbol{\theta}}^{[l+1]} \circ \boldsymbol{g}_{\boldsymbol{\theta}}^{[l+1]} \\
&= \begin{bmatrix} \boldsymbol{z}_{\boldsymbol{\theta}}^{[l]} \\ 0 \end{bmatrix} + \begin{bmatrix} \boldsymbol{0}_{m_{l+1}\times(s-1)} \\ -\alpha(\boldsymbol{z}_{\boldsymbol{\theta}}^{[l]})_s \\ \boldsymbol{0}_{m_{l+1}\times(m_l-s)} \\ \alpha(\boldsymbol{z}_{\boldsymbol{\theta}}^{[l]})_s \end{bmatrix} \\
&= \left[(\boldsymbol{z}_{\boldsymbol{\theta}}^{[l]})^\intercal_{[1:s-1]}, (1-\alpha)(\boldsymbol{z}_{\boldsymbol{\theta}}^{[l]})_s, (\boldsymbol{z}_{\boldsymbol{\theta}}^{[l]})^\intercal_{[s+1:m_l]}, \alpha(\boldsymbol{z}_{\boldsymbol{\theta}}^{[l]})_s \right]^\intercal.
\end{aligned} \tag{4}$$

Finally,

$$\begin{aligned}
\boldsymbol{z}_{\boldsymbol{\theta}'}^{[l-1]} &= \left[(\boldsymbol{W}^{[l]})^\intercal, (\boldsymbol{W}^{[l]})^\intercal_{s,[1:m_{l-1}]} \right] \left(\begin{bmatrix} \boldsymbol{z}_{\boldsymbol{\theta}}^{[l]} \\ 0 \end{bmatrix} + \begin{bmatrix} \boldsymbol{0}_{m_{l+1}\times(s-1)} \\ -\alpha(\boldsymbol{z}_{\boldsymbol{\theta}}^{[l]})_s \\ \boldsymbol{0}_{m_{l+1}\times(m_l-s)} \\ \alpha(\boldsymbol{z}_{\boldsymbol{\theta}}^{[l]})_s \end{bmatrix} \right) \circ \begin{bmatrix} \boldsymbol{g}_{\boldsymbol{\theta}}^{[l]} \\ (\boldsymbol{g}_{\boldsymbol{\theta}}^{[l]})_s \end{bmatrix} \\
&= (\boldsymbol{W}^{[l]})^\intercal \boldsymbol{z}_{\boldsymbol{\theta}}^{[l]} \circ \boldsymbol{g}_{\boldsymbol{\theta}}^{[l]} + \boldsymbol{0}_{m_{l-1}\times 1} \\
&= \boldsymbol{z}_{\boldsymbol{\theta}}^{[l-1]}.
\end{aligned} \tag{5}$$

This with the recurrence relation again leads to $\boldsymbol{z}_{\boldsymbol{\theta}'}^{[l']} = \boldsymbol{z}_{\boldsymbol{\theta}}^{[l']}$ for all $l' \in [1 : l-1]$. □

Proposition (Proposition 1: one-step embedding preserves network properties). *Given a L-layer ($L \geq 2$) fully-connected neural network with width $(m_0, \ldots, m_L)$, for any network parameters $\boldsymbol{\theta} = (\boldsymbol{W}^{[1]}, \boldsymbol{b}^{[1]}, \cdots, \boldsymbol{W}^{[L]}, \boldsymbol{b}^{[L]})$ and for any $l \in [L-1]$, $s \in [m_l]$, the following network properties are preserved for $\boldsymbol{\theta}' = \mathcal{T}^{\alpha}_{l,s}(\boldsymbol{\theta})$:*

(i) output function is preserved: $f_{\boldsymbol{\theta}'}(\boldsymbol{x}) = f_{\boldsymbol{\theta}}(\boldsymbol{x})$ for all $\boldsymbol{x}$;

(ii) empirical risk is preserved: $R_S(\boldsymbol{\theta}') = R_S(\boldsymbol{\theta})$;

(iii) the sets of features are preserved: $\left\{ \left(\boldsymbol{f}_{\boldsymbol{\theta}'}^{[l]} \right)_i \right\}_{i\in[m_l+1]} = \left\{ \left(\boldsymbol{f}_{\boldsymbol{\theta}}^{[l]} \right)_i \right\}_{i\in[m_l]}$ and $\left\{ \left(\boldsymbol{f}_{\boldsymbol{\theta}'}^{[l']} \right)_i \right\}_{i\in[m_{l'}]} = \left\{ \left(\boldsymbol{f}_{\boldsymbol{\theta}}^{[l']} \right)_i \right\}_{i\in[m_{l'}]}$ for $l' \in [L]\backslash\{l\}$;

Proof. The properties (i)–(iii) are direct consequences of Lemma 1. □

Theorem (Theorem 1: criticality preserving). *Given a L-layer ($L \geq 2$) fully-connected neural network with width $(m_0, \ldots, m_L)$, for any network parameters $\boldsymbol{\theta} = (\boldsymbol{W}^{[1]}, \boldsymbol{b}^{[1]}, \cdots, \boldsymbol{W}^{[L]}, \boldsymbol{b}^{[L]})$ and for any $l \in [L-1]$, $s \in [m_l]$, if $\nabla_{\boldsymbol{\theta}} R_S(\boldsymbol{\theta}) = \boldsymbol{0}$, then $\nabla_{\boldsymbol{\theta}} R_S(\boldsymbol{\theta}') = \boldsymbol{0}$.*

Proof. Gradient of loss with respect to network parameters of each layer can be computed from $\boldsymbol{F}$, $\boldsymbol{G}$, and $\boldsymbol{Z}$ as follows

$$\begin{aligned}
\nabla_{\boldsymbol{W}^{[l']}} R_S(\boldsymbol{\theta}) &= \nabla_{\boldsymbol{W}^{[l']}} \mathbb{E}_S \ell(\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{y}) = \mathbb{E}_S \left(\boldsymbol{z}_{\boldsymbol{\theta}}^{[l']} \circ \boldsymbol{g}_{\boldsymbol{\theta}}^{[l']} (\boldsymbol{f}_{\boldsymbol{\theta}}^{[l'-1]})^\intercal \right), \\
\nabla_{\boldsymbol{b}^{[l']}} R_S(\boldsymbol{\theta}) &= \nabla_{\boldsymbol{b}^{[l']}} \mathbb{E}_S \ell(\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{y}) = \mathbb{E}_S (\boldsymbol{z}_{\boldsymbol{\theta}}^{[l']} \circ \boldsymbol{g}_{\boldsymbol{\theta}}^{[l']}).
\end{aligned}$$

Then, by Lemma 1, we have $\nabla_{\boldsymbol{W}^{[l']}} R_S(\boldsymbol{\theta}') = \nabla_{\boldsymbol{W}^{[l']}} R_S(\boldsymbol{\theta}) = \boldsymbol{0}$ for $l' \neq l, l+1$ and $\nabla_{\boldsymbol{b}^{[l']}} R_S(\boldsymbol{\theta}') = \nabla_{\boldsymbol{b}^{[l']}} R_S(\boldsymbol{\theta}) = \boldsymbol{0}$ for $l' \neq l$. Also, for any $j \in [m_{l+1}]$,$k \in [m_l]$, since $(\boldsymbol{z}_{\boldsymbol{\theta}'}^{[l+1]})_j = (\boldsymbol{z}_{\boldsymbol{\theta}}^{[l+1]})_j$, $(\boldsymbol{g}_{\boldsymbol{\theta}'}^{[l+1]})_j = (\boldsymbol{g}_{\boldsymbol{\theta}}^{[l+1]})_j$, and $(\boldsymbol{f}_{\boldsymbol{\theta}'}^{[l]})_k = (\boldsymbol{f}_{\boldsymbol{\theta}}^{[l]})_k$, $(\boldsymbol{f}_{\boldsymbol{\theta}'}^{[l]})_{m_l+1} = (\boldsymbol{f}_{\boldsymbol{\theta}}^{[l]})_s$, we obtain

$$\begin{aligned}
\nabla_{\boldsymbol{W}^{[l+1]}_{j,k}} R_S(\boldsymbol{\theta}') &= \nabla_{\boldsymbol{W}^{[l+1]}_{j,k}} R_S(\boldsymbol{\theta}) = 0, \\
\nabla_{\boldsymbol{W}^{[l+1]}_{j,m_l+1}} R_S(\boldsymbol{\theta}') &= \nabla_{\boldsymbol{W}^{[l+1]}_{j,s}} R_S(\boldsymbol{\theta}) = 0.
\end{aligned}$$

Similarly, for any $j \in [m_l]\backslash\{s\}$,$k \in [m_{l-1}]$, we have

$$\begin{aligned}
\nabla_{\boldsymbol{W}_{j,k}^{[l]}} R_S(\boldsymbol{\theta}') &= \nabla_{\boldsymbol{W}_{j,k}^{[l]}} R_S(\boldsymbol{\theta}) = 0,\\
\nabla_{\boldsymbol{b}_{j}^{[l]}} R_S(\boldsymbol{\theta}') &= \nabla_{\boldsymbol{b}_{j}^{[l]}} R_S(\boldsymbol{\theta}) = 0,\\
\nabla_{\boldsymbol{W}_{s,k}^{[l]}} R_S(\boldsymbol{\theta}') &= (1-\alpha)\nabla_{\boldsymbol{W}_{s,k}^{[l]}} R_S(\boldsymbol{\theta}) = 0,\\
\nabla_{\boldsymbol{W}_{m_l+1,k}^{[l]}} R_S(\boldsymbol{\theta}') &= \alpha\nabla_{\boldsymbol{W}_{s,k}^{[l]}} R_S(\boldsymbol{\theta}) = 0,\\
\nabla_{\boldsymbol{b}_{s}^{[l]}} R_S(\boldsymbol{\theta}') &= (1-\alpha)\nabla_{\boldsymbol{b}_{s}^{[l]}} R_S(\boldsymbol{\theta}) = 0,\\
\nabla_{\boldsymbol{b}_{m_l+1}^{[l]}} R_S(\boldsymbol{\theta}') &= \alpha\nabla_{\boldsymbol{b}_{s}^{[l]}} R_S(\boldsymbol{\theta}) = 0.
\end{aligned}$$

Collecting all the above equalities, we have $\nabla_{\boldsymbol{\theta}} R_S(\boldsymbol{\theta}') = \mathbf{0}$. □

Lemma (**Lemma 2: increment of the degree of degeneracy**). *Given a L-layer ($L \geq 2$) fully-connected neural network with width $(m_0, \ldots, m_L)$, if there exists $l \in [L-1]$, $s \in [m_l]$, and a q-dimensional differential manifold $\mathcal{M}$ consisting of critical points of R_S such that for any $\boldsymbol{\theta} \in \mathcal{M}$, $\boldsymbol{W}_{[1:m_{l+1}],s}^{[l+1]} \neq \mathbf{0}$, then $\mathcal{M}' := \{\mathcal{T}_{l,s}^{\alpha}(\boldsymbol{\theta}) | \boldsymbol{\theta} \in \mathcal{M}, \alpha \in \mathbb{R}\}$ is a $(d+1)$-dimensional differential manifold consists of critical points for the corresponding L-layer fully-connected neural network with width $(m_0, \ldots, m_{l-1}, m_l + 1, m_{l+1}, \ldots, m_L)$.*

Proof. For any $\boldsymbol{\theta} \in \mathcal{M}$, let $\{\boldsymbol{e}_i(\boldsymbol{\theta})\}_{i=1}^{d}$ be a basis of its tangent space $T_{\boldsymbol{\theta}}\mathcal{M}$. Then for any $\alpha \in \mathbb{R}$, the tangent space of $\boldsymbol{\theta}' = \mathcal{T}_{l,s}^{\alpha}(\boldsymbol{\theta}) \in \mathcal{M}'$ is spanned by $\{\mathcal{T}_{l,s}(\boldsymbol{e}_1(\boldsymbol{\theta})), \cdots, \mathcal{T}_{l,s}(\boldsymbol{e}_d(\boldsymbol{\theta})), \mathcal{V}_{l,s}(\boldsymbol{\theta})\}$. Since $\mathcal{T}_{l,s}$ is linear and injective, $\{\mathcal{T}_{l,s}(\boldsymbol{e}_i(\boldsymbol{\theta}))\}_{i=1}^{d}$ is also a linearly independent set. Moreover, since $\boldsymbol{W}_{[1:m_{l+1}],m_l+1}^{[l+1]} = \mathbf{0}$ for any vector in parameter space applied with $\mathcal{T}_{l,s}$, then, $\{\mathcal{T}_{l,s}(\boldsymbol{e}_1(\boldsymbol{\theta})), \cdots, \mathcal{T}_{l,s}(\boldsymbol{e}_d(\boldsymbol{\theta})), \mathcal{V}_{l,s}(\boldsymbol{\theta})\}$ are independent if and only if $\mathcal{V}_{l,s}(\boldsymbol{\theta}) \neq \mathbf{0}$, i.e., $\boldsymbol{W}_{[1:m_{l+1}],s}^{[l+1]} \neq \mathbf{0}$. □

Remark 1. *The requirement that $\mathcal{M}$ is a q-dimensional differential manifold can be relaxed to that $\mathcal{M}$ is a q-dimensional topological manifold. In the latter case, $\mathcal{M}'$ is a $(d+1)$-dimensional topological manifold.*

Theorem (**Theorem 2: degeneracy of embedded critical points**). *Consider two L-layer ($L \geq 2$) fully-connected neural networks $\mathrm{NN}_A(\{m_l\}_{l=0}^{L})$ and $\mathrm{NN}_B(\{m_l'\}_{l=0}^{L})$ which is K-neuron wider than NN_A. Suppose that the critical point $\boldsymbol{\theta}_A = (\boldsymbol{W}^{[1]}, \boldsymbol{b}^{[1]}, \cdots, \boldsymbol{W}^{[L]}, \boldsymbol{b}^{[L]})$ satisfy $\boldsymbol{W}^{[l]} \neq \mathbf{0}$ for each layer $l \in [L]$. Then the parameters $\boldsymbol{\theta}_A$ of NN_A can be critically embedded to a K-dimensional critical affine subspace $\mathcal{M}_B = \{\boldsymbol{\theta}_B + \sum_{i=1}^{K} \alpha_i \boldsymbol{v}_i | \alpha_i \in \mathbb{R}\}$ of loss landscape of NN_B. Here $\boldsymbol{\theta}_B = (\prod_{i=1}^{K} \mathcal{T}_{l_i,s_i})(\boldsymbol{\theta}_A)$ and $\boldsymbol{v}_i = \mathcal{T}_{l_K,s_K} \cdots \mathcal{V}_{l_i,s_i} \cdots \mathcal{T}_{l_1,s_1} \boldsymbol{\theta}_A$.*

Proof. The assumption $\boldsymbol{W}^{[l]} \neq \mathbf{0}$, $l \in [L]$ implies the existence of non-silent neurons, i.e., existing $s \in [m_l]$ such that $\boldsymbol{W}_{[1:m_{l+1}],s}^{[l+1]} \neq \mathbf{0}$, for any $l \in [L-1]$ with $m_l' > m_l$.

In this proof, we misuse notation and denote $m_l = m_l(\boldsymbol{\theta})$ for the width of the l-th layer for any fully-connected neural network with parameters $\boldsymbol{\theta}$. For such a general network with parameters $\boldsymbol{\theta}$, we introduce the following operator. Given an index set J and for any $l \in [L]$, $s \in [m_l]$, we define

$$\mathcal{V}_{l,s,J}(\boldsymbol{\theta}) = \left(\mathbf{0}_{m_0 \times m_1}, \cdots, \left[\mathbf{0}_{m_{l+1} \times (s-1)}, -\sum_{j \in J} \boldsymbol{W}_{[1:m_{l+1}],j}^{[l+1]}, \mathbf{0}_{m_{l+1} \times (m_l - s)}, \sum_{j \in J} \boldsymbol{W}_{[1:m_{l+1}],j}^{[l+1]} \right], \cdots \right).$$

Clearly, $\mathcal{V}_{l,s,J} = \sum_{j\in J}\mathcal{V}_{l,s,\{j\}}$. If for all $j \in J$, $\boldsymbol{W}^{[l]}_{j,[1:m_{l-1}]} = \boldsymbol{W}^{[l]}_{s,[1:m_{l-1}]}$, $\boldsymbol{W}^{[l+1]}_{[1:m_{l+1}],j} = \beta_j \sum_{j'\in J}\boldsymbol{W}^{[l+1]}_{[1:m_{l+1}],j'}$ and $\sum_{j'\in J}\boldsymbol{W}^{[l+1]}_{[1:m_{l+1}],j'} \neq \mathbf{0}$, then for $\beta_s \neq 0$, we have

$$\begin{aligned}
\mathcal{T}^{\alpha}_{l,s,J}\boldsymbol{\theta} &= (\mathcal{T}_{l,s} + \alpha\mathcal{V}_{l,s,J})\boldsymbol{\theta} \\
&= (\mathcal{T}_{l,s} + \alpha\sum_{j\in J}\mathcal{V}_{l,s,j})\boldsymbol{\theta} \\
&= (\mathcal{T}_{l,s} + \alpha\sum_{j\in J}\frac{\beta_j}{\beta_s}\mathcal{V}_{l,s})\boldsymbol{\theta} \\
&= \mathcal{T}^{\alpha'}_{l,s}\boldsymbol{\theta},
\end{aligned}$$

where $\alpha' = \alpha\sum_{j\in J}\frac{\beta_j}{1-\beta_j}$, is simply the one-step critical embedding. We can extend this to the case of $\beta_s = 0$.

Then, for $J' = J \cup \{m_l + 1\}$, $l' = l$, $s' \in J$,

$$\begin{aligned}
\mathcal{V}_{l',s',J'}\mathcal{V}_{l,s,J} &= (\mathcal{V}_{l,s',J} + \mathcal{V}_{l,s',m_l+1})\mathcal{V}_{l,s,J} \\
&= \mathcal{V}_{l,s',J}\mathcal{V}_{l,s,J} + \mathcal{V}_{l,s',m_l+1}\mathcal{V}_{l,s,J} \\
&= \mathcal{V}_{l,s',s}\mathcal{V}_{l,s,J} + \mathcal{V}_{l,s',m_l+1}\mathcal{V}_{l,s,J} \\
&= \mathbf{0}.
\end{aligned}$$

In general, we have

$$\begin{aligned}
\mathcal{V}_{l',s',J'}\prod_{i=1}^{N}\mathcal{T}_{l'_i,s'_i}\mathcal{V}_{l,s,J} &= (\mathcal{V}_{l,s',J} + \mathcal{V}_{l,s',m_l+1})\prod_{i=1}^{N}\mathcal{T}_{l'_i,s'_i}\mathcal{V}_{l,s,J} \\
&= \mathcal{V}_{l,s',J}\prod_{i=1}^{N}\mathcal{T}_{l'_i,s'_i}\mathcal{V}_{l,s,J} + \mathcal{V}_{l,s',m_l+1}\prod_{i=1}^{N}\mathcal{T}_{l'_i,s'_i}\mathcal{V}_{l,s,J} \\
&= \mathcal{V}_{l,s',s}\prod_{i=1}^{N}\mathcal{T}_{l'_i,s'_i}\mathcal{V}_{l,s,J} + \mathcal{V}_{l,s',m_l+1}\prod_{i=1}^{N}\mathcal{T}_{l'_i,s'_i}\mathcal{V}_{l,s,J} \\
&= \mathbf{0}.
\end{aligned}$$

For $l' \neq l$ or $s' \notin J$, obviously we have $\mathcal{V}_{l',s',J'}\mathcal{V}_{l,s,J} = \mathbf{0}$ and $\mathcal{V}_{l',s',J'}\prod_{i=1}^{N}\mathcal{T}_{l'_i,s'_i}\mathcal{V}_{l,s,J} = \mathbf{0}$.

Now we are ready to prove the lemma. Let $J_i = \{s_i\} \cup \{m_l + \#\{i|l_i = l, i \in [j]\}|l_j = l, s_j = s, j \in [K]\}$, where $\#$ indicates number of elements in a set. Then

$$\begin{aligned}
\prod_{i=1}^{K}\mathcal{T}^{\alpha_i}_{l_i,s_i,J_i} &= \prod_{i=1}^{K}(\mathcal{T}_{l_i,s_i} + \alpha_i\mathcal{V}_{l_i,s_i,J_i}) \\
&= \prod_{i=1}^{K}\mathcal{T}_{l_i,s_i} + \sum_{i=1}^{K}\alpha_i\mathcal{T}_{l_K,s_K}\cdots\mathcal{V}_{l_i,s_i,J_i}\cdots\mathcal{T}_{l_1,s_1}, \\
&= \prod_{i=1}^{K}\mathcal{T}_{l_i,s_i} + \sum_{i=1}^{K}\alpha_i\mathcal{T}_{l_K,s_K}\cdots\mathcal{V}_{l_i,s_i}\cdots\mathcal{T}_{l_1,s_1},
\end{aligned}$$

which is a critical embedding for any $[\alpha_i]_{i=1}^{K} \in \mathbb{R}^K$. This completes the proof.

□

A.1 Trivial critical transforms

In general, neuron-index permutation among the same layer is a trivial criticality invariant transform because of the layer-wise intrinsic symmetry of DNN models. Therefore, any critical point/manifold

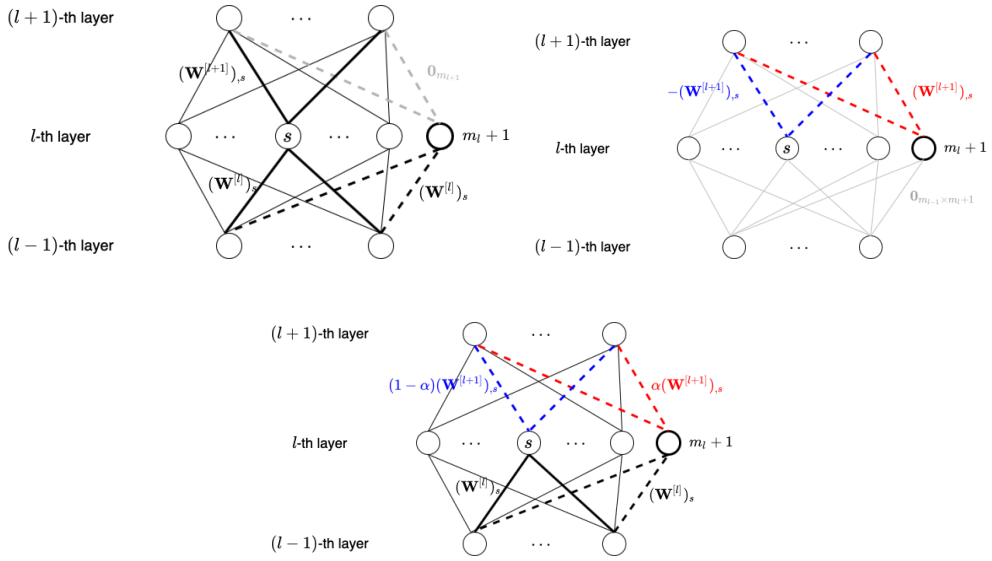

Figure S1: Illustration of $\mathcal{T}_{l,s}$, $\mathcal{V}_{l,s}$, and $\mathcal{T}^{\alpha}_{l,s}$.

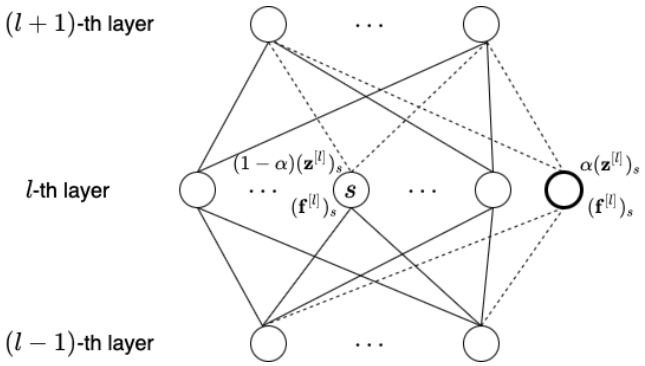

Figure S2: Illustration of $\boldsymbol{F}$ and $\boldsymbol{Z}$

may result in multiple "mirror" critical points/manifolds of the loss landscape through all possible permutations. However, this transform does not inform about the degeneracy of critical points/manifolds.

For any p-homogeneous activation function σ i.e., $\sigma(\beta x) = \beta^p \sigma(x)$ for any $\beta > 0$ and $x \in \mathbb{R}$, we define for any $l \in [L-1]$, $s \in [m_l]$ the following scaling transform $\boldsymbol{\theta}' = \mathcal{S}^{\beta}_{l,s}(\boldsymbol{\theta})$ $(\beta \neq 0)$ such that $\boldsymbol{W}'^{[l+1]}_{[1:m_{l+1}],s} = \frac{1}{\beta^p}\boldsymbol{W}^{[l+1]}_{[1:m_{l+1}],s}$ and $\boldsymbol{W}'^{[l]}_{s,[1:m_{l-1}]} = \beta \boldsymbol{W}^{[l]}_{s,[1:m_{l-1}]}, \boldsymbol{b}'^{[l]}_s = \beta \boldsymbol{b}^{[l]}_s$, and all the other entries remain the same. Clearly, this transform is also a critical transform. Moreover, it informs about one more degenerate dimension for each neuron with $\left\|\boldsymbol{W}^{[l+1]}_{[1:m_{l+1}],s}\right\|_2 \left\|\left(\boldsymbol{W}^{[l]\intercal}_{s,[1:m_{l-1}]}, \boldsymbol{b}^{[l]}_s\right)^{\intercal}\right\|_2 \neq 0$. This critical scaling transform is trivial in a sense that it is an obvious result of the cross-layer scaling preserving intrinsic to each DNN of homogeneous activation function, not relevant to cross-width landscape similarity between DNNs we focus on.

B Details of experiments

For the 1D fitting experiments (Figs. 1, 3(a), 4), we use tanh as the activation function, mean squared error (MSE) as the loss function. We use the full-batch gradient descent with learning rate 0.005 to train NNs for 300000 epochs. The initial distribution of all parameters follows a normal distribution with a mean of 0 and a variance of $\frac{1}{m^3}$.

For the iris classification experiment (Fig. 3(b)), we use sigmoid as the activation function, MSE as the loss function. We use the default Adam optimizer of full batch with learning rate 0.02 to train for 500000 epochs. The initial distribution of all parameters follows a normal distribution with mean 0 and variance $\frac{1}{m^6}$.

For the experiment of MNIST classification (Fig. 5), we use ReLU as the activation function, MSE as the loss function. We also use the default Adam optimizer of full batch with learning rate 0.00003 to train for 100000 epochs. The initial distribution of all parameters follows a normal distribution with mean 0 and variance $\frac{1}{m^6}$.

To obtain the empirical diagram in Fig. 4, we run 200 trials each for width-1, width-2 and width-3 tanh NNs with variance of initial parameters $\frac{1}{m^3}$ $(m = 1, 2, 3)$ for 300000 epochs. Then we find all parameters with gradient less than 10^{-10}, which we define as empirical critical points, throughout the training in total 600 trajectories. Next, we cluster them based on their loss values, output functions, input parameters of neurons and only 4 different cases arises after excluding the trivial case of constant zero output. Their output functions are shown in the figure.

Remark that, although Figs. 1 and 5 are case studies each based on a random trial, similar phenomenon can be easily observed as long as the initialization variance is properly small, i.e., far from the linear/kernel/NTK regime.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 6.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 5.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] In the material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 3.2
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] the provider information Will be shown in Acknowledgement.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No] The MNIST datatset is well known.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]