Directly Constructing Low-Dimensional Solution Subspaces in Deep Neural Networks

Yusuf Kalyoncuoglu
Department of Computer Science
RWTH Aachen University
yusuf.kalyoncuoglu@rwth-aachen.de

December 28, 2025

Abstract

While it is well-established that the weight matrices and feature manifolds of deep neural networks exhibit a low Intrinsic Dimension (ID), current state-of-the-art models still rely on massive high-dimensional widths. This redundancy is not required for representation, but is strictly necessary to solve the non-convex *optimization search problem*—finding a global minimum, which remains intractable for compact networks.

In this work, we propose a constructive approach to bypass this optimization bottleneck. By decoupling the solution geometry from the ambient search space, we empirically demonstrate across ResNet-50, ViT, and BERT that the classification head can be compressed by even huge factors of 16 with negligible performance degradation.

This motivates *Subspace-Native Distillation* as a novel paradigm: by defining the target directly in this constructed subspace, we provide a stable geometric coordinate system for student models, potentially allowing them to circumvent the high-dimensional search problem entirely and realize the vision of "Train Big, Deploy Small."

1 Introduction

The scaling laws of deep learning dictate that larger models with higher-dimensional widths generally yield better performance (1). Consequently, state-of-the-art architectures such as Transformers in NLP and Vision operate with embedding dimensions ranging from $d = 768$ (Vision-Transformer, (2)) to over $d = 12000$ (GPT-3, (3)). While this over-parameterization aids the optimization process by smoothing the loss landscape (4), it raises a fundamental question regarding inference efficiency and representation geometry: *Does achieving state-of-the-art accuracy fundamentally require high-dimensional representations, or is this dimensionality merely a temporary aid for the optimization process?*

Empirical evidence points to the latter, suggesting that the redundancy is structural, rooted in the spectral properties of the weight matrices themselves. Analyzing the Empirical Spectral Density (ESD) of deep networks, (5) identified a phenomenon of “Heavy-Tailed Self-Regularization.” They demonstrated that well-trained weight matrices do not utilize their full algebraic rank; instead, the information concentrates along a small number of dominant singular vectors, while the vast majority of dimensions constitute noise or “bulk” components (5).

This spectral sparsity implies that the network scales the signal only along very few critical directions. This observation holds across architectures, including Convolutional Neural Networks, where the spectral properties have been extensively studied (6). Specifically, it has been shown that the singular values of convolution operators decay sharply, allowing for aggressive low-rank approximations without retraining (7).

It is precisely this low-rank geometry that enables the success of modern compression and adaptation techniques. Methods like Singular Value Decomposition (SVD) (8) and Low-Rank Adaptation

(LoRA) (9) work effectively because they exploit the fact that weight matrices and updates reside in low-dimensional subspaces. Essentially, these methods acknowledge that the model scales along very few critical directions.

This leads to a paradox: If the final solution lives in a low-dimensional subspace, why do we not simply design and train smaller backbones from scratch? The *Lottery Ticket Hypothesis* (10) provides a critical insight into this: it posits that large, dense networks contain sparse subnetworks—"winning tickets"—that are capable of matching the full model's accuracy when trained in isolation. The high-dimensional parameter space effectively serves as a vast combinatorial pool, significantly increasing the probability that at least one such optimal subnetwork exists at initialization and can be successfully uncovered by gradient descent.

This is further formalized by *Neural Tangent Kernel* (NTK) theory (11), which shows that in the infinite-width limit, deep networks transition into a "lazy training" regime. In this state, the network's evolution is governed by a static kernel, effectively linearizing the optimization landscape and guaranteeing convergence to a global minimum (12). Small networks, however, operate far from this regime; their optimization landscapes remain highly non-convex and chaotic, making it significantly harder to find the global minimum despite having the theoretical *representational capacity* to hold the solution. Essentially, high dimensionality provides the **optimization capacity** required to make the search tractable.

Knowledge Distillation (KD) attempts to bridge this gap by using a teacher to guide the student's optimization trajectory (13) (14). However, standard KD methods typically force the student to mimic the teacher's full high-dimensional output logits or feature maps (15). This imposes an unnecessarily hard constraint: the student must approximate the entire ambient space of the teacher—including its noise and redundant bulk components. This "capacity gap" can hinder effective transfer if the student struggles to model the teacher's high-dimensional complexity (16).

Consequently, the high dimensionality of SOTA models serves primarily to facilitate this search. We propose a direct validation of this premise by asking: Can we directly construct the solution in a much smaller space?

By "constructing the solution subspace", we refer to the explicit transformation of the backbone's final latent activations $h \in \mathbb{R}^d$ into a fixed, lower-dimensional space $\tilde{h} \in \mathbb{R}^k$ via a data-independent projection Φ. This $\mathbb{R}^k$ serves as the definitive space where the classification task is solved, proving that the high-dimensional ambient space d is not required for linear separability.

Proving the existence of such a robust, constructible subspace would imply that the complexity lies in the *process*, not the *result*. By isolating this target geometry, we effectively transform the daunting search problem into a clearer regression task. If the exact low-dimensional "destination" is known, a student backbone can be optimized to construct this specific subspace directly.

To validate this hypothesis, we propose a constructive diagnostic approach. Instead of searching for an optimal subspace via data-dependent methods, we utilize fixed, data-independent *Johnson-Lindenstrauss (JL)* projections (17). This allows us to rigorously test whether the solution manifold is linearly separable in a random low-dimensional basis, independent of the backbone's optimization trajectory.

Our contributions are as follows:

- We perform a systematic evaluation of fixed subspace classification across three distinct architectures and modalities: BERT (18) on MNLI (19), ViT (2) on ImageNet-100 (20), and ResNet-50 (21) on CIFAR-100 (22).

- We systematically map the performance of the **solution space** across multiple reduction factors. Our results show that even under extreme compression of up to 16x (e.g., $2048 \rightarrow 128$ or $768 \rightarrow 64$), the model maintains negligible performance degradation ($\approx 1\%$). This consistent stability across various dimensions proves that the intrinsic solution is not only low-dimensional but also remarkably robust to random geometric contraction.

- We establish the existence of a robust "solution subspace". Based on this, we propose *Subspace-Native Distillation* as a future paradigm, where student models are trained to directly target this low-dimensional manifold, bypassing the redundancy of high-dimensional teacher representations.

2 Related Work

Our work sits at the intersection of geometric analysis of deep representations, the theory of neural collapse, and dimensionality reduction via random projections.

2.1 Intrinsic Dimensionality of Deep Representations

Despite the massive parameter count of modern networks, growing evidence suggests that the effective "operational dimension" is surprisingly low. Ansuini et al. (24) analyzed the intrinsic dimension (ID) of intermediate layers, showing that while the ID is high in early layers, it drastically collapses in the final layers, forming a "curved" to "flat" transition. Similarly, Pope et al. (25) demonstrated that the intrinsic dimension of image manifolds sets a fundamental limit on the generalization capability of the model. While these works focus on *measuring* the dimension using estimators (e.g., nearest-neighbor distances), our work focuses on *constructing* a tangible subspace that suffices for classification.

2.2 Neural Collapse and Spectral Properties

The geometric structure of the final classification layer has been formalized under the framework of "Neural Collapse" (23). They observed that as training progresses towards zero training error, the within-class variability of features vanishes, and class means converge to the vertices of a Simplex Equiangular Tight Frame (ETF). This implies that the optimal separating hyperplanes lie in a low-rank subspace determined solely by the number of classes, rather than the ambient dimension d. Complementing this, (5) applied Random Matrix Theory to show that well-trained weight matrices exhibit heavy-tailed spectral densities, where information is concentrated in a few dominant singular values. Our use of random projections empirically validates these theories: the fact that JLL projections preserve separability confirms that the signal-to-noise ratio is dominated by these few heavy-tailed components.

2.3 Random Projections and Model Compression

The Johnson-Lindenstrauss Lemma (17) has historically been used in machine learning for speeding up kernel machines (27) or for compressed sensing. In the context of Deep Learning, Li et al. (26) utilized random projections to measure the "intrinsic dimension of the objective landscape," showing that optimization can occur in a randomly oriented low-dimensional subspace. However, most compression techniques, such as PCA-based reduction or autoencoders (28) (29), rely on *data-dependent* bases. In contrast, our approach utilizes *oblivious* (data-independent) projections. By proving that a fixed random basis performs on par with the full model, we show that the representation is robust enough to survive aggressive geometric distortion, paving the way for simpler, subspace-native architectures.

3 Methodology

3.1 Theoretical Motivation: Manifold Collapse

Let $f_\theta : \mathcal{X} \to \mathbb{R}^d$ denote the pre-trained Teacher backbone (e.g., BERT or ResNet) with parameters θ. While the ambient output dimension d is large (e.g., 768 or 2048), Ansuini et al. (24) demonstrated that the local intrinsic dimension of the data manifold $\mathcal{M} = f_\theta(\mathcal{X})$ drops significantly in the final layers. Furthermore, Papyan et al. (23) identified the *Neural Collapse* phenomenon, where intra-class variability vanishes and class means converge to a low-rank structure.

These findings imply that the high-dimensional vector $h \in \mathbb{R}^d$ is sparse in information content. The effective solution resides in a subspace $\mathcal{S} \subset \mathbb{R}^d$ with dimension $k \ll d$. Our goal is to isolate $\mathcal{S}$

directly. If the manifold is indeed flattened and low-rank, a random linear projection should be sufficient to capture its geometry without the need for learned, non-linear autoencoders.

3.2 Constructing the Solution via Johnson-Lindenstrauss

To construct this solution subspace, we employ *Random Projections* rooted in the Johnson-Lindenstrauss Lemma (JLL) (17). We choose JLL specifically because it provides a theoretical guarantee for preserving the Euclidean geometry of a low-dimensional manifold embedded in a high-dimensional space, irrespective of the specific basis vectors.

The JLL states that for a set of points in high-dimensional space, a random projection into $k = O(\epsilon^{-2} \log N)$ dimensions preserves pairwise distances within a factor of $(1 \pm \epsilon)$. We define our projection operator $\Phi : \mathbb{R}^d \to \mathbb{R}^k$ via a random matrix $R \in \mathbb{R}^{k \times d}$, where each entry is drawn independently from a standard normal distribution:

$$R_{ij} \sim \mathcal{N}(0, 1) \tag{1}$$

The low-dimensional solution vector $\tilde{h} \in \mathbb{R}^k$ is obtained by projecting the Teacher's output:

$$\tilde{h} = \frac{1}{\sqrt{k}} R h \tag{2}$$

The scaling factor $\frac{1}{\sqrt{k}}$ ensures the preservation of the expected norm. Crucially, R is generated at initialization and remains **frozen**. This ensures that we are strictly assessing the intrinsic separability of the Teacher's representation. By using an oblivious (data-independent) projection, we prove that the separability is a property of the data geometry itself, not an artifact of a learned compression layer.

3.3 Subspace Classification as Target Construction

We treat the projected vector $\tilde{h}$ as the input to our "Solution Subspace." To validate that this subspace contains all necessary semantic information, we train a lightweight linear classifier $g_W : \mathbb{R}^k \to \mathbb{R}^C$ directly on $\tilde{h}$.

The training objective minimizes the Cross-Entropy loss over the dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$:

$$\min_{W,b} \sum_{(x,y) \in \mathcal{D}} \mathcal{L}_{CE} \left(\text{softmax}(W \tilde{h}_i + b), y_i \right) \tag{3}$$

where $\tilde{h}_i = \Phi(f_\theta(x_i))$.

During this phase, the Teacher backbone θ is strictly frozen. Consequently, g_W acts as a proxy for the minimal complexity required to solve the task. If g_W achieves performance parity with the full-dimensional model, it confirms that we have successfully constructed a valid solution subspace. This $\tilde{h}$ can subsequently be viewed as the ideal "ground truth" signal for future student models, decoupling the solution from the high-dimensional optimization path of the teacher.

3.4 Dimensionality Reduction Factors

We vary the target dimension k across a wide spectrum. For ResNet-50 ($d = 2048$), we evaluate subspaces of width $k \in \{1024, 512, 256, 128\}$, corresponding to compression factors up to 16x. For BERT and ViT ($d = 768$), we evaluate $k \in \{512, 256, 128, 64\}$, reaching compression factors of 12x.

3.5 Experimental Implementation Details

To ensure the reproducibility and robustness of our findings, we implemented the proposed framework across three distinct modalities using a unified three-phase protocol: (1) Full Fine-Tuning (to establish an upper bound), (2) Frozen Linear Probing (to establish the full-dimensional baseline), and (3) Subspace

Classification (our method). All experiments were conducted with a fixed random seed ($s = 42$) to ensure deterministic behavior of the JLL projection matrices.

1. Vision (CNN): ResNet-50 on CIFAR-100 We utilized a ResNet-50 backbone pre-trained on ImageNet-1k. The feature vector was extracted from the final pooling layer ($d = 2048$) before the fully connected layer.

- *Preprocessing:* We employed data augmentation to prevent overfitting on the small 32×32 CIFAR images, including Random Crop, Horizontal Flip, Color Jitter, and *Cutout* regularization (30) (1 hole, max size 8).
- *Optimization:* The model was optimized using SGD with momentum (0.9). For the Subspace phase, we trained for 5 epochs with a learning rate of $1e^{-2}$ and weight decay of $5e^{-4}$.

2. NLP (Transformer): BERT-base on MNLI We employed the `bert-base-uncased` model ($d = 768$) evaluated on the GLUE/MNLI matched validation set. The input consists of premise-hypothesis pairs with a maximum sequence length of 128 tokens.

- *Feature Extraction:* We utilized the `pooler_output` (the embedding of the [CLS] token processed by a dense layer and Tanh activation) as the input h for our projections.
- *Optimization:* We used the AdamW optimizer (31). While Full Fine-Tuning required a conservative learning rate ($2e^{-5}$), the Subspace Classification phase allowed for a higher learning rate ($1e^{-3}$) to rapidly converge the linear head on the frozen features.

3. Vision (Transformer): ViT-B/16 on ImageNet-100 We utilized a Vision Transformer (ViT-B/16) pre-trained on ImageNet-21k ($d = 768$) and fine-tuned on ImageNet-100, a standard subset of 100 classes.

- *Preprocessing:* Images were resized to 224×224. Similar to ResNet, we applied *Cutout* (max size 32) during training to encourage robust feature learning. The [CLS] token of the last hidden state served as the feature vector.
- *Optimization:* Training was performed with AdamW. As with BERT, we utilized a learning rate of $1e^{-3}$ for the subspace classifiers to effectively regress the target decision boundaries within 3 epochs.

Table 1: **Hyperparameter Configuration.** Specific settings for the Subspace Construction phase (Frozen Backbone). Note the use of higher learning rates for Transformers to adapt the linear head to the fixed manifold.

Model	Optimizer	LR (Head)	Weight Decay	Epochs	Batch Size
ResNet-50	SGD	1×10^{-2}	5×10^{-4}	5	128
BERT-base	AdamW	1×10^{-3}	1×10^{-2}	3	32
ViT-B/16	AdamW	1×10^{-3}	1×10^{-4}	3	32

4 Empirical Evaluation

In this section, we present the experimental results of constructing solution subspaces via Johnson-Lindenstrauss projections. We evaluate our hypothesis across three distinct modalities and architectures to ensure the universality of the phenomenon.

4.1 Main Results

Table 2 summarizes the classification accuracy across all three tasks. We report the raw accuracy and the relative difference (Δ) compared to the full-dimensional Frozen Baseline.

Table 2: **Subspace Classification Results.** Comparison of Full Fine-Tuning, Frozen Baseline (Full Dim), and JLL Subspace Projections. Δ denotes the performance difference relative to the Frozen Baseline. Note the stability of accuracy even at high compression rates (e.g., 12x).

Model / Task	**Method**	**Dim (k)**	**Ratio**	**Accuracy**	**Δ Baseline**
ResNet-50	Full Fine-Tuning	2048	1x	79.95%	-2.45%
(CIFAR-100)	**Frozen Baseline**	**2048**	**1x**	**82.40%**	**–**
	JLL Projection	1024	2x	82.29%	-0.11%
	JLL Projection	512	4x	81.96%	-0.44%
	JLL Projection	256	8x	81.69%	-0.71%
	JLL Projection	128	16x	81.19%	-1.21%
ViT-B/16	Full Fine-Tuning	768	1x	93.84%	-0.18%
(ImageNet-100)	**Frozen Baseline**	**768**	**1x**	**94.02%**	**–**
	JLL Projection	512	1.5x	93.88%	-0.14%
	JLL Projection	**256**	**3x**	**94.32%**	**+0.30%**
	JLL Projection	128	6x	93.90%	-0.12%
	JLL Projection	64	12x	93.86%	-0.16%
BERT-base	Full Fine-Tuning	768	1x	83.69%	-0.05%
(MNLI)	**Frozen Baseline**	**768**	**1x**	**83.74%**	**–**
	JLL Projection	512	1.5x	83.60%	-0.14%
	JLL Projection	256	3x	83.58%	-0.16%
	JLL Projection	128	6x	83.36%	-0.38%
	JLL Projection	64	12x	83.70%	-0.04%

4.2 Analysis of Results

Robustness of the Solution Subspace: Across all architectures, the performance drop is minimal even under aggressive compression. For BERT (and ViT), reducing the dimension from 768 to 64 (a 12x reduction) results in a negligible accuracy loss. Similarly, ResNet-50 retains over 98.5% of its baseline performance when projected down to 128 dimensions. This confirms our hypothesis that the intrinsic solution manifold is significantly smaller than the ambient space allocated by the architecture.

5 Discussion and Conclusion

5.1 Theoretical Implications: The Flat Solution Manifold

Our findings provide constructive evidence for the "Neural Flattening" hypothesis. While prior work measured the intrinsic dimension via local gradients or manifold estimation (24), our use of a global, data-independent linear projection offers a stronger guarantee: the solution manifold is not only low-dimensional locally but is also globally "well-behaved" (flat and linearly separable).

The logic is as follows: If the class manifolds were highly curved or entangled in the high-dimensional space, a random linear projection would inevitably lead to significant class overlap and performance collapse (violating the margin requirements for JLL). The fact that this collapse does not occur implies that the backbone effectively linearizes the problem into a subspace $k \ll d$. The feature extractor acts as a "flattening engine," rendering the complex input data into a geometry so simple that even a random basis is sufficient to distinguish the classes.

5.2 Future Direction: Subspace-Native Distillation

Our experiments have empirically confirmed the existence of a robust solution subspace that captures nearly the entire performance of the teacher despite being a fraction of the size. This finding fundamentally challenges current Knowledge Distillation paradigms. Since we have proven that the relevant signal exists in the projected subspace $\tilde{h} = Rh$, we propose **Subspace-Native Distillation**. In this framework, the student is trained to directly construct the solution in the low-dimensional manifold defined by the fixed matrix R. The loss function shifts to:

$$\mathcal{L}_{subspace} = ||h_{student} - Rh_{teacher}||^2 \tag{4}$$

By targeting the intrinsic dimension directly, we hypothesize that the student can bypass the high-dimensional search problem entirely. The backbone of the student only needs enough capacity to map inputs to the k-dimensional target, potentially allowing for extreme parameter reduction while maintaining the teacher's accuracy. This paves the way for a generation of "subspace-native" models that are optimized for the solution, not the search.

5.3 Conclusion

While high-dimensional width is crucial for solving the non-convex "search problem" during training, the final solution resides in a much simpler, low-rank subspace. This distinction provides the theoretical and empirical foundation for *Subspace-Native Distillation*. By shifting the distillation target from the noisy ambient space to this verified solution manifold, we can potentially train student models that are efficient by design rather than by approximation.

Ultimately, our framework validates a clear architectural philosophy for the future of efficient deep learning: **Train Big** to leverage the optimization benefits of high-dimensional scaffolding, but **Deploy Small** by explicitly extracting and targeting the intrinsic solution that remains.

References

[1] Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., ... & Amodei, D. (2020). Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361.*

[2] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations (ICLR).*

[3] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems (NeurIPS)*, 33, 1877–1901.

[4] Li, H., Xu, Z., Taylor, G., Studer, C., & Goldstein, T. (2018). Visualizing the Loss Landscape of Neural Nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 31.

[5] Martin, C. H., & Mahoney, M. W. (2021). Implicit self-regularization in deep neural networks: Evidence from random matrix theory and implications for learning. *Journal of Machine Learning Research*, 22(165), 1-73.

[6] Sedghi, H., Gupta, V., & Long, P. M. (2018). The singular values of convolutional layers. *International Conference on Learning Representations (ICLR).*

[7] Idelbayev, Y., & Carreira-Perpinan, M. A. (2020). Low-rank compression of neural networks: Learning the rank of each layer. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).*

[8] Denton, E. L., Zaremba, W., Bruna, J., LeCun, Y., & Fergus, R. (2014). Exploiting linear structure within convolutional networks for efficient evaluation. *Advances in Neural Information Processing Systems (NeurIPS)*, 27.

[9] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... & Chen, W. (2022). LoRA: Low-Rank Adaptation of Large Language Models. *International Conference on Learning Representations (ICLR).*

[10] Frankle, J., & Carbin, M. (2019). The lottery ticket hypothesis: Finding sparse, trainable neural networks. *International Conference on Learning Representations (ICLR).*

[11] Jacot, A., Gabriel, F., & Hongler, C. (2018). Neural tangent kernel: Convergence and generalization in neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 31.

[12] Lee, J., Xiao, L., Schoenholz, S., Bahri, Y., Novak, R., Sohl-Dickstein, J., & Pennington, J. (2019). Wide neural networks of any depth evolve as linear models under gradient descent. *Advances in Neural Information Processing Systems (NeurIPS)*, 32.

[13] Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the Knowledge in a Neural Network. *arXiv preprint arXiv:1503.02531.*

[14] Phuong, M., & Lampert, C. (2019). Towards understanding knowledge distillation. *International Conference on Machine Learning (ICML).*

[15] Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., & Bengio, Y. (2015). FitNets: Hints for thin deep nets. *International Conference on Learning Representations (ICLR).*

[16] Cho, J. H., & Hariharan, B. (2019). On the efficacy of knowledge distillation. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV).*

[17] Johnson, W. B., & Lindenstrauss, J. (1984). Extensions of Lipschitz mappings into a Hilbert space. *Contemporary mathematics*, 26(189-206), 1.

[18] Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186.

[19] Williams, A., Nangia, N., and Bowman, S. (2018). A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1112–1122.

[20] Tian, Y., Krishnan, D., and Isola, P. (2020). Contrastive Multiview Coding. In *European Conference on Computer Vision (ECCV)*, pages 776–794.

[21] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

[22] Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. *Technical Report, University of Toronto.*

[23] Papyan, V., Han, X. Y., and Donoho, D. L. (2020). Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40), 24652–24663.

[24] Ansuini, A., Laio, A., Macke, J. H., and Zoccolan, D. (2019). Intrinsic dimension of data representations in deep neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 32.

[25] Pope, P., Zhu, C., Abdelkader, A., Goldblum, M., and Goldstein, T. (2021). The Intrinsic Dimension of Images and Its Impact on Learning. In *International Conference on Learning Representations (ICLR)*.

[26] Li, C., Farkhoor, H., Liu, R., and Yosinski, J. (2018). Measuring the Intrinsic Dimension of Objective Landscapes. In *International Conference on Learning Representations (ICLR)*.

[27] Rahimi, A. and Recht, B. (2007). Random features for large-scale kernel machines. *Advances in neural information processing systems (NeurIPS)*, 20.

[28] Pearson, K. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11), 559–572.

[29] Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786), 504–507.

[30] DeVries, T., & Taylor, G. W. (2017). Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*.

[31] Loshchilov, I., & Hutter, F. (2019). Decoupled weight decay regularization. *International Conference on Learning Representations (ICLR)*.