# GEOMETRIC PROPERTIES OF NEURAL MULTIVARIATE REGRESSION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Neural multivariate regression underpins a wide range of domains such as control, robotics, and finance, yet the geometry of its learned representations remains poorly characterized. While neural collapse has been shown to benefit generalization in classification, we find that analogous collapse in regression consistently degrades performance. To explain this contrast, we analyze models through the lens of intrinsic dimension. Across control tasks and synthetic datasets, we estimate the intrinsic dimension of last-layer features ($ID_H$) and compare it with that of the regression targets ($ID_Y$). Collapsed models exhibit $ID_H < ID_Y$, leading to over-compression and poor generalization, whereas non-collapsed models typically maintain $ID_H > ID_Y$. For the non-collapsed models, performance with respect to $ID_H$ depends on the data quantity and noise levels. From these observations, we identify two regimes—over-compressed and under-compressed—that determine when expanding or reducing feature dimensionality improves performance. Our results provide new geometric insights into neural regression and suggest practical strategies for enhancing generalization.

## 1 INTRODUCTION

Neural multivariate regression has emerged as a cornerstone of modern machine learning, powering a wide spectrum of applications where the outputs are continuous and vector-valued. In imitation learning for autonomous driving, regression models predict control actions such as speed and steering angle from human driving demonstrations. In robotics, they enable agents to replicate expert trajectories. In finance, regression underlies predictive analytics ranging from risk estimation to stock price forecasting. Finally, reinforcement learning employs regression to approximate value functions, with targets derived from Monte Carlo or bootstrapped returns. The ubiquity of regression across these domains underscores the importance of a principled understanding of the representational geometry learned by neural networks in multivariate regression tasks.

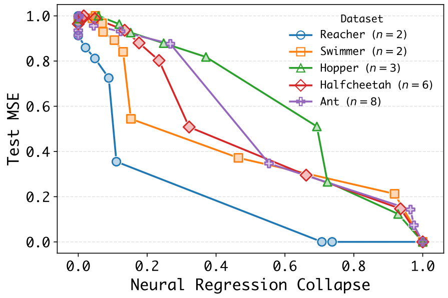


Figure 1: Neural Regression Collapse typically correlates with high Test MSE. The smaller the NRC value, the closer the features lie to the $n$-dimensional subspace.

In this work, we empirically investigate the *geometric structure of neural multivariate regression*, with an emphasis on the geometry of last-layer feature vectors. Prior efforts have largely framed this problem through the lens of *neural collapse*. In classification, Neural Collapse (NC) describes the emergence of a highly symmetric configuration: last-layer features converge to the vertices of a Simplex Equiangular Tight Frame (ETF), aligned with the classifier weights (Papyan et al., 2020). In regression, by contrast, Neural Regression Collapse (NRC) manifests as the concentration of last-layer features within a linear subspace spanned by the top $n$ principal components of the last-layer feature matrix, where $n$ is the number of target variates. Since $n$ is typically much smaller than the feature dimension, regression collapse implies a major reduction in representational degrees of freedom (Andriopoulos et al., 2024).

In this paper we first make a key empirical observation: *In contrast with classification, collapsed regression models consistently exhibit degraded generalization as compared to their non-collapsed counterparts.* Figure 1 illustrates this phenomenon, showing high values of test MSE for models with highly collapsed features (low values of the NRC metric) for five robotic locomotion tasks. Existing theoretical and empirical treatments of regression collapse, including the work of (Andriopoulos et al., 2024), do not account for this degradation. This raises a central open question: ***Why does neural collapse hinder generalization in multivariate regression, in contrast to its beneficial role in classification?***

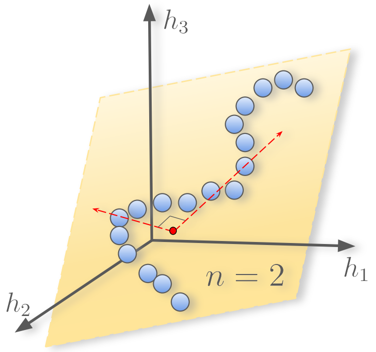


Figure 2: When the target dimension is $n = 2$, the collapsed features (blue points) lie close to a subspace (yellow plane) spanned by the first 2 principal components (red arrows) of the last-layer features. Moreover, the collapsed features lie in a non-linear manifold of smaller dimension than $n$.

We address this question by employing *intrinsic dimension (ID)*, which as compared with the methodology of neural regression collapse, is a more refined tool for analyzing the geometry of multivariate regression. The intrinsic dimension of a dataset quantifies the effective dimensionality of the manifold in which the data lies. While intrinsic dimension has been previously studied in the context of neural classification (Ansuini et al., 2019), to the best of our knowledge this is the first work to analyze neural multivariate regression from this perspective. As shown in Figure 2 and studied in the paper, intrinsic dimension can capture nonlinearities that the PCA approach of NRC cannot. Employing the 2-NN global estimator for intrinsic dimension (Facco et al., 2017), we conduct a systematic investigation of collapsed and non-collapsed models across diverse regression tasks, including simulated robotic locomotion in continuous control environments and synthetic regression tasks constructed from standard image datasets.

Our findings reveal that the intrinsic dimension of the regression targets, denoted $ID_Y$, is a critical threshold for understanding model geometry. We denote the intrinsic dimension of the last-layer features as $ID_H$, the intrinsic dimension of the inputs as $ID_X$, and the intrinsic dimension of the output predictions as $ID_P$. We observe that in collapsed models, typically $ID_H < ID_Y$, whereas in non-collapsed models, typically $ID_H > ID_Y$. This systematic discrepancy explains the poor generalization of collapsed regression models: *the degradation stems from an over-compression of learned representations.* Due to this over-compression, it is not possible to recover the target manifold from the lower-dimensional feature manifold.

From this perspective, we identify two distinct regimes of generalization in neural regression. In the *over-compressed regime* ($ID_H < ID_Y$), generalization can be improved when the intrinsic dimension of last-layer features is increased, for example by altering network architectures or regularization parameters. In the *under-compressed regime* ($ID_H > ID_Y$), the opposite holds when the training set is small or noisy: reducing intrinsic dimension yields gains. Together, these results not only explain the detrimental role of collapse in regression but also suggest strategies for improving generalization in practice.

This paper makes the following contributions:

- We provide, to the best of our knowledge, the first systematic investigation of neural multivariate regression through the lens of intrinsic dimension.
- We empirically demonstrate that regression collapse corresponds to a regime where the intrinsic dimension of last-layer features falls below that of the targets, explaining its negative impact on generalization.
- We show that the relative relationship between $ID_Y$ and $ID_H$ identifies two regimes — over-compressed and under-compressed — and the conditions under which adjusting feature dimension improves generalization performance.
- Our results yield a more refined geometric understanding of regression representations and suggest practical strategies for improving generalization in applied regression tasks.

## 2 Related work

**NC under varied settings on classification.** The phenomenon of neural collapse was first empirically observed by (Papyan et al., 2020), who demonstrated its emergence during TPT in deep neural network models for classification tasks. Building on this empirical finding, researchers have developed theoretical frameworks to analyze NC such as the Unconstrained Feature Model (UFM) (Mixon et al., 2020) and the layer-peeled model (Fang et al., 2021). Using these models, numerous studies have demonstrated that NC provably occurs under diverse conditions (Han et al., 2021; Tirer & Bruna, 2022; Yaras et al., 2022; Zhou et al., 2022a;b; Zhu et al., 2021) and using various loss functions such as label smoothing (Guo et al., 2024). See also (Hong & Ling, 2023; Thrampoulidis et al., 2022; Yang et al., 2022).

**NC beyond single-label classification.** Recent research has extended the principles of NC beyond its original single-label classification setting. (Li et al., 2023a) demonstrated that in multi-label classification, embeddings reside within the linear span of their label means. (Andriopoulos et al., 2024) generalized NC to neural multivariate regression, formalizing it as Neural Regression Collapse (NRC). Concurrently, (Ma et al., 2025) showed that NC also emerges in deep ordinal regression, analyzing it through the UFM framework. In large-scale language models, (Wu & Papyan, 2024) identified a “linguistic collapse". (Súkeník et al., 2025) proved that NC represents the globally optimal configuration in modern deep regularized architectures, including ResNets and transformers.

**Intrinsic dimension in deep neural networks.** Several works investigate the intrinsic dimension of data manifolds and representations in deep neural networks (Denil et al., 2013; LeCun et al., 1989). Classical methods estimate intrinsic dimension from local neighborhoods (Allegra et al., 2020; Amsaleg et al., 2015; Facco et al., 2017; Levina & Bickel, 2004), which has been extended to neural settings. For instance, (Ma et al., 2018a) show that local intrinsic dimension (LID) can distinguish adversarial from natural image data. More recently, (Yin et al., 2024) focused on per-sample LID to identify when LLMs produce untruthful outputs.

A parallel line of research uses tools from topological data analysis to study neural networks. Some works analyze the final trained network by constructing topological invariants from the layer weights such as Neural Persistence (Rieck et al., 2018), which can distinguish between models that overfit or generalize well. Others analyze the underlying graph structure of networks (Corneanu et al., 2019; 2020). While often empirical, these approaches provide a novel perspective on network properties. More recent work (Birdal et al., 2021) has begun to place these topological methods on a firmer theoretical foundation using statistical persistent homology.

Beyond empirical estimations, intrinsic dimension has been studied as a measure of model complexity. Recent approaches analyze the degrees of freedom in parameter space (Gao & Jojic, 2016; Janson et al., 2015), compressibility via pruning (Blier & Ollivier, 2018), and intrinsic dimension (Ansuini et al., 2019; Li et al., 2018; Ma et al., 2018b; Pope et al., 2021). Compression-based generalization bounds (Arora et al., 2018; Barsbey et al., 2021; Hsu et al., 2021; Suzuki et al., 2018; 2019) have shown that networks that can be represented in a lower-dimensional space exhibit lower generalization error. See also (Simsekli et al., 2020; Birdal et al., 2021; Zhu et al., 2018).

## 3 Background and key metrics

We consider the multivariate regression problem with $M$ training examples $\{(\mathbf{x}_i, \mathbf{y}_i), i = 1, ..., M\}$, where each input $\mathbf{x}_i$ belongs to $\mathbb{R}^D$ and each target vector $\mathbf{y}_i$ belongs to $\mathbb{R}^n$. For the regression task, the neural network takes as input an example $\mathbf{x} \in \mathbb{R}^D$ and produces an ouput $\mathbf{y} = f(\mathbf{x}) \in \mathbb{R}^n$. For most neural networks, including those used in this paper, this mapping takes the form

$$f_{\theta,\mathbf{W},\mathbf{b}}(\mathbf{x}) = \mathbf{W}\mathbf{h}_\theta(\mathbf{x}) + \mathbf{b},$$

where $\mathbf{h}_\theta(\cdot) : \mathbb{R}^D \to \mathbb{R}^d$ is the non-linear feature extractor consisting of several non-linear layers, $\mathbf{W}$ is a $n \times d$ matrix representing the final linear layer in the model, and $\mathbf{b} \in \mathbb{R}^n$ is the bias vector. The parameters $\theta, \mathbf{W}, \mathbf{b}$ are all trainable.

We typically train the DNN using gradient descent to minimize the regularized L2 loss:

$$\min_{\theta,\mathbf{W},\mathbf{b}} \frac{1}{2M} \sum_{i=1}^{M} ||f_{\theta,\mathbf{W},\mathbf{b}}(\mathbf{x}_i) - \mathbf{y}_i||_2^2 + \frac{\lambda_{WD}}{2}(||\theta||_2^2 + ||\mathbf{W}||_F^2),$$

where $||\cdot||_2$ and $||\cdot||_F$ denote the $L_2$-norm and the Frobenius norm, respectively. As commonly done in practice, in our experiments we set all the regularization parameters to the same value, which we refer to as the weight decay parameter $\lambda_{WD}$.

To characterize the geometric properties of last-layer representations of neural networks in regression tasks, we consider two central metrics: the NRC1 metric; the 2-Nearest Neighbor (2-NN) intrinsic dimension estimator.

### 3.1 The Neural Regression Collapse: NRC1 metric

Neural collapse in classification describes the convergence of last-layer features to a simplex-like structure. In regression, neural collapse is defined by the extent to which the last-layer feature vectors collapse to a subspace spanned by their top principal components (PCs).

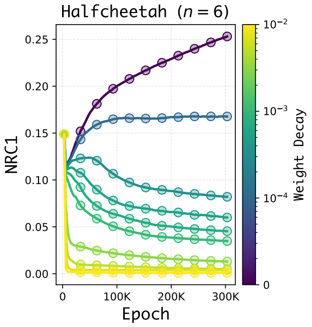


Figure 3: NRC1 decreases with stronger weight decay, leading to model collapse.

Let $\mathbf{h}_i := \mathbf{h}_\theta(\mathbf{x}_i)$ be the feature vector associated with example $\mathbf{x}_i$, $i = 1, \ldots, M$. Further let $\widetilde{\mathbf{h}}_i$ be the normalized feature vector, that is, $\widetilde{\mathbf{h}}_i := (\mathbf{h}_i - \bar{\mathbf{h}}) \cdot ||\mathbf{h}_i - \bar{\mathbf{h}}||^{-1}$ where $\bar{\mathbf{h}} := M^{-1}\sum_{i=1}^{M} \mathbf{h}_i$. For any $p \times q$ matrix $\mathbf{C}$ and any $p$-dimensional vector $\mathbf{v}$, let proj$(\mathbf{v}|\mathbf{C})$ denote the projection of $\mathbf{v}$ onto the subspace spanned by the columns of $\mathbf{C}$. Let $\mathbf{H}_{\text{PCA}}$ be the $d \times n$ matrix with the columns consisting of the first $n$ PCs of $\mathbf{H}$.

The NRC1 metric is defined as

$$\text{NRC1} := \frac{1}{M}\sum_{i=1}^{M} ||\widetilde{\mathbf{h}}_i - \text{proj}(\widetilde{\mathbf{h}}_i|\mathbf{H}_{\text{PCA}})||_2^2,$$

which measures the extent to which the last-layer features concentrate around their top $n$ principal components. A model is considered collapsed if NRC1 is small, indicating that the features lie almost entirely within an $n$-dimensional subspace. Non-collapsed models have higher values of NRC1, differing from those of collapsed models by orders of magnitude. Figure 3 investigates NRC1 for values of the weight decay parameter $\lambda_{WD}$. We see that when $\lambda_{WD}$ is zero or small, there is no neural regression collapse; but if we increase the weight decay, the NRC1 geometric structure quickly emerges during training. In Appendix G, we also empirically show how dropout regularization and model depth influence NRC1 and $ID_H$ with zero or mild weight decay.

### 3.2 Intrinsic dimension via 2-NN estimation

To uncover the finer geometric structure of the learned features, beyond what linear methods like PCA reveal, we turn into intrinsic dimension — the minimal number of degrees of freedom needed to describe the data without significant information loss. To estimate the intrinsic dimension, we use the 2-NN estimator, introduced by (Facco et al., 2017) and further explored in deep learning contexts by (Ansuini et al., 2019; Pope et al., 2021) to study the properties of the internal representations of CNNs. The 2-NN estimator is notable for its minimal assumptions. Unlike the estimates in (Levina & Bickel, 2004; Ceruti et al., 2014), the 2-NN estimator only requires the dataset to be locally uniform in density, where locally means in the range of the second neighbor.

For a given point, let $r_1$ and $r_2$ denote the distances to its first and second nearest neighbors, respectively; define the ratio $\mu := r_2/r_1$. Under the assumption of locally uniform sampling, the cumulative distribution of the ratio above follows a Pareto distribution with parameter $d$: $F(\mu) = 1 - \mu^{-d}$ for $\mu \geq 1$. The intrinsic dimension $d$ is then estimated by linear regression. Details are provided in the Appendix C.

In addition to the minimal neighborhood assumption, the estimator has other key advantages. It provides stable estimates for intrinsic dimension below $\sim 20$, even for modest sample sizes and non-uniform densities. Moreover, it is scale-aware, meaning that by sub-sampling the dataset, 2-NN can discriminate between “soft" (relevant) and “noisy" dimensions, see Fig. 3 in (Facco et al., 2017).

## 4 Datasets

We perform experiments on robotic locomotion and vision-based datasets in the main body, which are summarized in Table 5. In Appendix H, we also examine four more challenging tasks with varying sizes, increased intrinsic dimensions, and visual inputs.

Table 1: Overview of datasets employed in our experiments.

| **Dataset** | **Data Size** | **Input Type** | **Input Dim** $(D)$ | **Input ID** $(ID_X)$ | **Target Dim** $(n)$ | **Target ID** $(ID_Y)$ |
|---|---|---|---|---|---|---|
| Swimmer | 20,000 | raw state | 8 | 4.03 | 2 | 1.34 |
| Reacher | 20,000 | raw state | 11 | 3.80 | 2 | 1.83 |
| Hopper | 20,000 | raw state | 11 | 4.51 | 3 | 2.91 |
| Halfcheetah | 20,000 | raw state | 17 | 6.76 | 6 | 5.29 |
| Ant | 20,000 | raw state | 111 | 7.19 | 8 | 7.29 |
| MNIST | 50,000 | Grayscale image | $28 \times 28$ | 12.76 | 25 | 8.02 |
| CIFAR-10 | 50,000 | RGB image | $32\times 32\times 3$ | 27.20 | 10 | 9.51 |

**MuJoCo locomotion**: MuJoCo (Todorov et al., 2012; Brockman et al., 2016; Towers et al., 2023) is a physics simulator that is widely used as a continuous-control benchmark in reinforcement learning. Following Andriopoulos et al. (2024), we adopt the Reacher, Swimmer, and Hopper datasets. Moreover, the Halfcheetah and Ant datasets of a higher target dimension are included from the standard D4RL benchmark (Fu et al., 2020). Each dataset consists of expert demonstration trajectories, where inputs are robotic proprioceptive sensing $(\mathbf{x}_i)$ and targets are the corresponding actions $(\mathbf{y}_i)$. The states encode joint positions, angles, velocities, and angular velocities, while the actions correspond to the torques applied to each joint. We subsample a portion of the expert trajectories to create low-data and high-data settings consisting of 1,000 and 20,000 samples, respectively. In Appendix A.1, we discuss more about the MuJoCo environments.

**Vision-based regression**: We create two regression tasks using the MNIST and CIFAR-10 image datasets. The goal is to produce one set of regression targets that is low-noise and another that contains significant task-irrelevant information. We provide a detailed formulation of noisy-target tasks in Appendix A.2. For both tasks, all target vectors are normalized to have zero mean and unit variance.

*MNIST Regression*: This task is designed to be low-noise. First, we train a standard CNN on the MNIST classification task until it achieves over 99% accuracy. We then use this highly accurate model as a fixed feature extractor. For each input image $\mathbf{x}$, we take the 128-dimensional vector from the network's final hidden layer and project it down to a 25-dimensional target vector $\mathbf{y}$ using a fixed, random matrix. The estimated intrinsic dimension of these targets is 8.02.

*CIFAR-10 Regression*: This task is constructed to include a higher degree of noise in its targets. We use a ResNet-18 model, pretrained on ImageNet, to extract features from CIFAR-10 images. Importantly, this model is not fine-tuned on the CIFAR-10 dataset. This mismatch ensures the extracted features contain information not specific to the CIFAR-10 images, creating noisy targets. These features are then projected down to 10-dimensional target vectors $\mathbf{y}$ using a fixed, random matrix. The estimated intrinsic dimension of these targets is approximately 9.51.

## 5 Intrinsic Dimension versus NRC

The NRC1 metric measures the degree to which the features collapse to an $n$-dimensional linear subspace, where $n$ is the dimension of the targets. As shown in Figure 3, a small amount of regularization often suffices for such collapse to occur. The NRC1 metric, however, does not provide insight into whether the features collapse into lower-dimensional non-linear manifolds. To explore this issue, we measure the intrinsic dimension of the last-layer features via the 2-NN estimator.

Figure 4 presents scatter plots for intrinsic dimension versus NRC1 for four MuJoCo datasets. These plots provide the following insights:

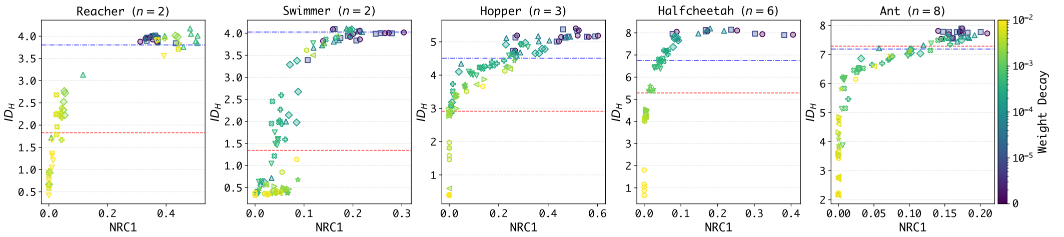


Figure 4: Relationship between NRC1 and intrinsic dimension of the last-layer features. Dots correspond to models trained with different architectures and weight decay parameters, with the colors denoting the degree of weight decay. The horizontal red dashed line is drawn at $ID_Y$.

- The critical value $ID_Y$, the intrinsic dimension of the targets, denoted by the horizontal dashed-red lines in Figure 4, is always below $n$, the dimension of the targets. Depending on the dataset, it can be significantly below.
- Highly collapsed models, i.e., those with small NRC1 values, learn last-layer features that lie on manifolds with intrinsic dimension below or in the vicinity of $ID_Y$, that is, $ID_H \lesssim ID_Y < n$. Thus, *for collapsed models, the last-layer features lie on a non-linear manifold that is within a linear subspace of dimension $n$.* These models have a wide range of $ID_H$ values, but are often clustered on a nearly vertical line below $ID_Y$. So, although the NRC1 metric is useful in understanding the linear-subspace structure of the last-layer features in collapsed models, it is inadequate at uncovering this more refined geometric structure.
- In contrast, for non-collapsed models, i.e., models with higher NRC1 values, the last-layer features satisfy $ID_H > ID_Y$. Thus, *for non-collapsed models, the last-layer features lie on a manifold with intrinsic dimension higher than the intrinsic dimension of the targets.* Furthermore, there is (approximately) a monotonic increasing relationship between NRC1 and $ID_H$. Thus, in this region, in terms of qualitative behavior, NRC1 and intrinsic dimension are interchangeable.

Thus, the 2-NN estimator has several advantages over the NRC1 metric, including uncovering a critical soft threshold $ID_Y$ corresponding to two NRC regimes, and also quantifying the degree of collapse (that is, $ID_H$) for all ranges of NRC1. For the remainder of this paper, we will therefore focus on intrinsic dimension.

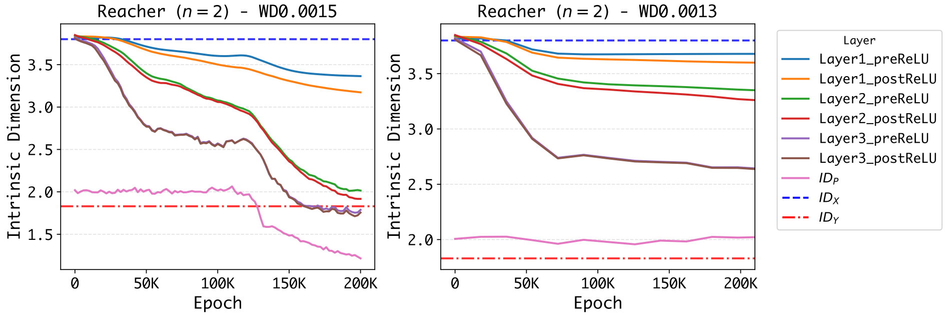


Figure 5: Intrinsic dimension of input, output, and hidden layers over training epochs for a collapsed (left) and a non-collapsed model (right) for the Reacher dataset. Each subfigure shows the evolution of intrinsic dimension across layers with blue, orange dashed and pink lines denoting the intrinsic dimension of inputs, targets, and predicted outputs, respectively.

To further understand the behavior of collapsed models and their counterparts, we track the evolution of the intrinsic dimension throughout training. Figure 5 provides illustrative examples for a collapsed and a non-collapsed model. Additional training curves are provided in the Appendix. From Figure 5, we have the following insights:

- For both the collapsed and non-collapsed models, during training, the intrinsic dimension of the last-layer features invariably decreases monotonically until convergence.

- For the collapsed model, the deeper the layer in the network, the lower the intrinsic dimension at the end of training. ReLU activations cause a mild reduction in intrinsic dimension in comparison with the reduction in intrinsic dimension between consecutive layers (ignoring ReLU). Notably, the final intrinsic dimension of the output layer, which gives the actual vector-valued predictions, can be significantly lower than $ID_H$.
- For non-collapsed models, we usually see — but not always (see Appendix B) — $ID_H$ decrease monotonically as we move from shallow to deep layers. Furthermore, we observe that during training the intrinsic dimension of the output layer hugs the intrinsic dimension of the targets. Thus, tracking the intrinsic dimension of the output layer provides yet another criterion for discriminating between collapsed and non-collapsed models; see Appendix E.

## 6 Intrinsic dimension and generalization

Having now investigated the relationship between neural regression collapse and intrinsic dimension, we now examine what insights intrinsic dimension can provide about generalization. Among other issues, we will explore why generalization error increases as $ID_H$ (and hence as NRC1) decreases, as seen in Figure 1. As we discuss in more detail at the end of this section, this property is in contrast with classification, for which performance typically improves when neural collapse becomes stronger.

Table 2: Key Takeaways for Generalization.

| Regime | ID | Typical behavior |
|---|---|---|
| Over-compressed | $\mathrm{ID}_H < \mathrm{ID}_Y$ | Underfitting with large train and test MSE |
| Balanced | $\mathrm{ID}_H \approx \mathrm{ID}_Y$ | Sweet spot in low-data and noisy tasks |
| Under-compressed | $\mathrm{ID}_H \gg \mathrm{ID}_Y$ | Benign overfitting with enough low-noise data |

Figure 6 shows the relationship between $ID_H$ and both training and test MSE for six datasets: four MuJoCo datasets, the CIFAR-10 dataset, and the MNIST dataset. (The corresponding figures for the remaining MuJoCo datasets are in the Appendix.) Figure 6 also provides the *generalization gap* which is defined to be the test MSE minus the train MSE.

**Train MSE decreases when $\mathbf{ID_H}$ increases.** This property is clearly visible in left column of Figure 6. To explain this, from Figures 3 and 4 we know stronger regularization reduces $ID_H$. From Theorems 4.1 and 4.3 in (Andriopoulos et al., 2024), we also know stronger regularization reduces the dimension of the linear subspace containing the feature manifold. Thus, by reducing $ID_H$, the trained features tend to get squashed onto a lower-dimensional and more curved manifold, similar to the "crowding problem" described by Maaten & Hinton (2008). A global linear layer $\mathbf{W}$ — which only performs rotation, scaling, and shearing — cannot "unbend" such a manifold. Thus as $ID_H$ decreases, it becomes more difficult for $\mathbf{WH} + \mathbf{b}$ to accurately match $\mathbf{Y}$ (which lies on its own curved manifold), explaining why train MSE decreases when $ID_H$ increases.

**Test MSE with respect to $\mathbf{ID_H}$ behaves differently according to its relationship to $\mathbf{ID_Y}$.** This can be seen in the middle column of Figure 6. There are fundamental differences between collapsed and non-collapsed models:

- $(ID_H < ID_Y)$: In this regime, the model's features are confined to a manifold whose intrinsic dimension is lower than that of the targets. This *over-compression* means the last-layer features lack information essential for reconstructing the full target manifold; see Section 6.1 for a theoretical explanation of this claim. This, in turn, leads to poor performance on both train and test data. The generalization gap is small not because the performance is good, but because the model fails for both seen and unseen data. In this regime, generalization can be improved when the intrinsic dimension of last-layer features is increased, for example, by altering the network architecture or the regularization parameters. We can now answer the question posed in the Introduction: Why does neural collapse hinder generalization in multivariate regression (as observed in Figure 1)? The explanation simply follows from $(i)$ the monotonic relationship between NRC1 and $ID_H$ and $(ii)$ the reconstruction loss that arises when $(ID_H < ID_Y)$, as just described.
- $(ID_H \geq ID_Y)$: Here we distinguish between two cases:

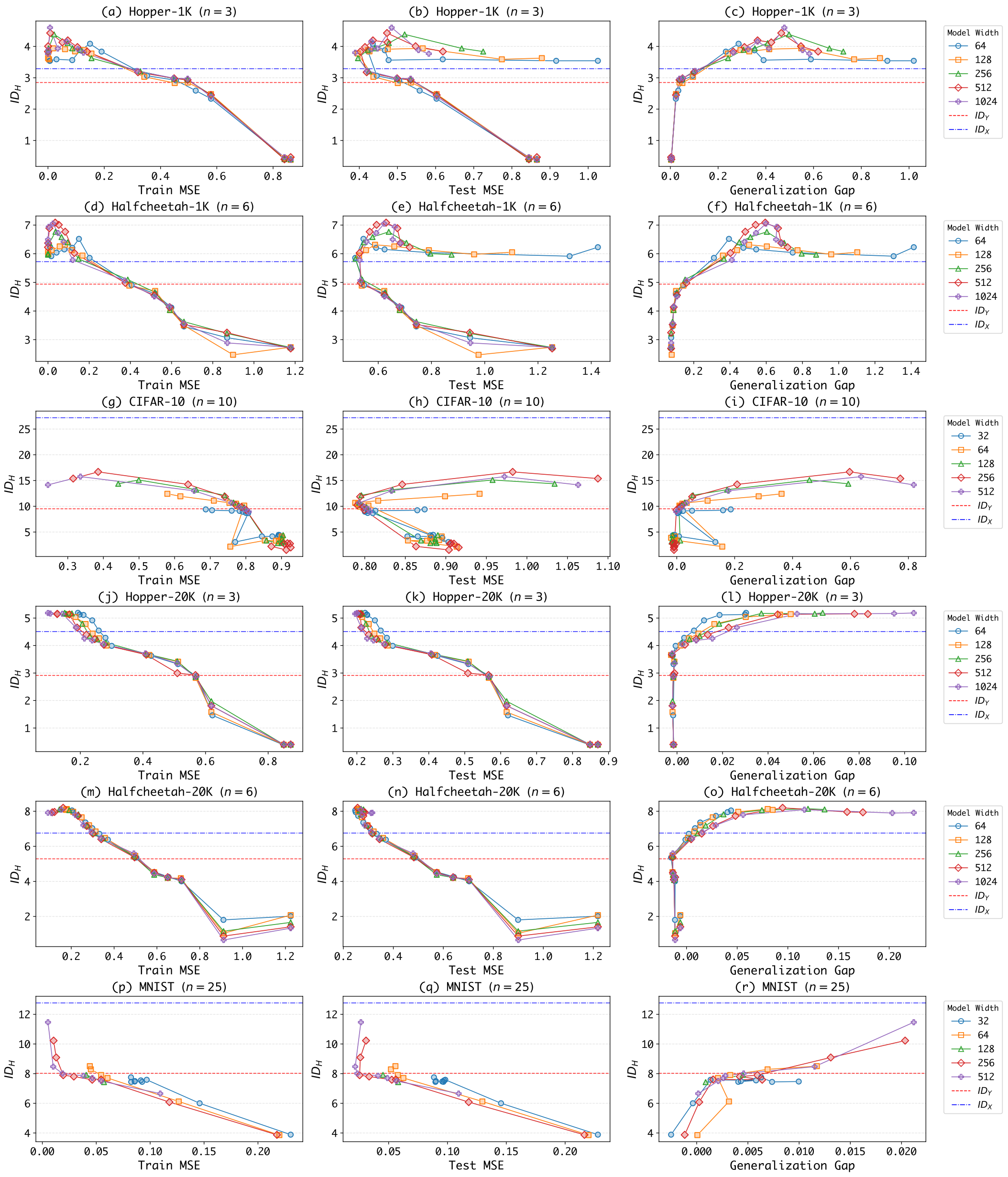


Figure 6: Generalization ability and Intrinsic Dimension for all datasets.

*(i) Low-data tasks and noisy-target tasks.* In this case (Figs. 6(b), (e), (h)), the test MSE scatter plots exhibit surprising U-shaped curves relative to $ID_H$, with minimum error occurring near $ID_H \simeq ID_Y$. Why does test MSE increase with $ID_H$ when $ID_H > ID_Y$? To explain this, we note that when the amount of training data is small, or when the targets are noisy, $f_\theta$ learns a feature manifold whose intrinsic dimension is higher than that of the true feature manifold because of the stronger negative effect of outliers. In these cases, the extra dimensions in the feature manifold are being used to predict training sample-specific noise, leading to overfitting the training set. This overfitting is exacerbated when regularization is reduced, or equivalently, when $ID_H$ is increased, leading to higher test MSE (Fig. 6). Thus in the low data and high noise regimes with $ID_H \gg ID_Y$, generalization can be improved when the intrinsic dimension of last-layer features is decreased.

*(ii) High-data tasks and low-noise tasks.* In this case, test MSE follows the same trend as train MSE, decreasing monotonically with $ID_H$ (Figs. 6 (k),(n),(q)). To explain this, we note that with a large amount of training data and low target noise, $f_\theta$ can fit the training data closely while

maintaining smoothmness to avoid overfitting, and consequently the manifold for $\mathbf{H}_{train}$ is similar to the manifold for $\mathbf{H}_{test}$. Table 2 provides the key takeaways concerning generalization and intrinsic dimension.

### 6.1 Mathematical Argument for Unavoidable Error in Collapsed Models

We now present two theoretical results that explain the fundamental limitations of collapsed models. First, we explain why weight decay causes neural regression collapse. Second, we show why collapsed models cannot achieve perfect reconstruction of the target manifold.

**Why weight decay causes collapse:** Using the Unconstrained Feature Model (UFM) framework employed in prior work on neural regression collapse (Andriopoulos et al., 2024), we analyze the effect of weight decay on solution structure in Appendix F. Theorem 4 shows that under the UFM framework, as weight decay approaches zero while maintaining a fixed ratio between feature and weight regularization, the learned features converge to the minimum-norm solution of the unregularized problem, which lies entirely within an $n$-dimensional subspace. This theoretical result demonstrates that weight decay fundamentally biases optimization toward collapsed solutions by eliminating the $(d-n)$-dimensional null space component of the features. While the UFM framework predicts collapse from arbitrarily small regularization, in practice we observe that stronger weight decay values are needed to induce collapse in actual neural networks, as shown in Figure 3.

**Why collapsed models fail to generalize:** We now provide a result from differential geometry showing that a smooth map (including a linear map $\mathbf{W}$) from a lower-dimensional manifold to a higher-dimensional one cannot be surjective, that is, it cannot cover all the points in the target manifold. The proof follows directly from Sard's Theorem and is provided in Appendix D.

**Theorem 1.** *Let $\mathcal{M}$ be a smooth $m$-dimensional manifold and $\mathcal{N}$ be a smooth $n$-dimensional manifold, with $m < n$. A smooth map $g : \mathcal{M} \to \mathcal{N}$ cannot be surjective, i.e., $g(\mathcal{M}) \neq \mathcal{N}$.*

This theorem provides the geometric foundation for understanding the failure of collapsed models. In our regression context, the learned features $\{\mathbf{h}_\theta(\mathbf{x})\}$ form a feature manifold $\mathcal{M}_H$ of dimension $m = ID_H$, while the targets $\{\mathbf{y}\}$ lie on a target manifold $\mathcal{N}_Y$ of dimension $n = ID_Y$. The final layer of the network constitutes a smooth map from the feature manifold to the target space.

When a model is in the over-compressed regime ($ID_H < ID_Y$), the theorem's condition ($m < n$) is met. The direct consequence is that this smooth map cannot be surjective. This means the image of the feature manifold—the set of all possible predictions the model can generate—is a proper subset of the target manifold. Geometrically, there will always be points on the target manifold that lie outside the model's predictive reach. A perfect reconstruction is therefore impossible, as the model is fundamentally incapable of generating the full range of target data, leading to an unavoidable error.

### 6.2 Comparison with Classification

Previous work on manifold learning for neural classification has demonstrated that the intrinsic dimension of the last hidden layer is positively correlated with generalization ability. In particular, models achieving lower intrinsic dimension in the penultimate layer were found to exhibit superior test accuracy, with the lowest intrinsic dimension-model attaining the highest top-5 accuracy, see Section 3.2 and Figure 4 in (Ansuini et al., 2019). Additionally, (Papyan et al., 2020) connect neural collapse to robust decision boundaries, (Galanti et al., 2021) demonstrate that collapse patterns improve few-shot and transfer learning, and (Li et al., 2022) show the degree of collapse in downstream representations strongly predicts transfer accuracy, Complementing these empirical results, there are also theoretical results showing the benefits of neural collapse for classification (Gao et al., 2023; Wang & Palmer, 2023; Hui et al., 2022).

In regression, however, our findings indicate a more nuanced picture. We demonstrated the existence of a “soft" threshold at $ID_Y$, which delineates distinct generalization regimes. In the under-compressed regime with low-data tasks and high-noise tasks, reducing $ID_H$ improves generalization, consistent with the monotonic complexity-performance paradigm observed in classification. However, in the over-compressed regime and in the under-compressed regime with high-data tasks and low-noise tasks, the opposite holds: increasing $ID_H$ improves generalization, a phenomenon absent in classification tasks. Thus, in regression, generalization performance depends non-monotonically on the relationship between the learned feature manifold and the intrinsic dimension of the targets.

## 7 Limitations and Future Work

While our work provides new geometric insights into neural multivariate regression through intrinsic dimension analysis, some limitations remain. Although we provide some theoretical results explaining why weight decay causes collapse (Appendix F) and why collapsed models often fail (Appendix D), a complete theoretical characterization of the relationship between intrinsic dimension and generalization is not yet available. Additionally, the 2-NN estimator we employ provides reliable estimates for intrinsic dimensions below approximately 20. For extremely high-dimensional target spaces or feature representations, alternative estimation methods may be necessary. Finally, our practical guidelines rely on adjusting standard hyperparameters (weight decay, model depth, dropout) to indirectly control $ID_H$. A more principled approach would be to optimize $ID_H$ during training; however, the 2-NN estimator is non-differentiable and cannot be optimized via backpropagation.

Several promising directions emerge from our findings. Deriving generalization bounds that incorporate intrinsic dimension would provide a rigorous theoretical foundation for the empirical relationships we observe. Theoretical frameworks beyond the UFM could offer additional perspectives on how network architecture, training dynamics, and regularization jointly determine the intrinsic dimension of learned representations. Differentiable intrinsic dimension estimation remains largely underexplored; developing robust, efficient and scalable differentiable estimators that enable direct control of $ID_H$ during training via backpropagation represents an important research direction. Such advances would allow practitioners to explicitly target desired intrinsic dimensions rather than adjusting hyperparameters indirectly, and would broaden the applicability of our analysis to higher-dimensional settings. Finally, exploring whether intrinsic dimension provides similar insights for other learning paradigms, such as generative modeling, reinforcement learning, and multi-task learning, could offer a unifying geometric perspective across domains.

## 8 Conclusion

In this paper, we provided a systematic geometric analysis of neural multivariate regression, highlighting a fundamental contrast with classification. Using intrinsic dimension, we showed that regression collapse corresponds to an over-compressed regime where the feature manifold has lower intrinsic dimension than the target manifold ($ID_H < ID_Y$), leading to consistently poor generalization. In contrast, non-collapsed models typically satisfy $ID_H \geq ID_Y$, with generalization behavior governed by whether the task is low-data/noisy or high-data/low-noise. These results establish intrinsic dimension as a principled diagnostic for understanding when collapse hinders regression performance.

Our findings yield two main contributions. First, they explain why collapse, beneficial in classification, is detrimental in regression: over-compression discards essential information needed to reconstruct target manifolds. Second, they provide practical guidelines: increasing $ID_H$ improves generalization in the over-compressed regime, while reducing $ID_H$ can help in noisy or low-data settings. Together, these insights refine the geometric understanding of regression representations and suggest principled strategies for improving generalization in applied multivariate regression tasks.

## Reproducibility Statement

To ensure the reproducibility of our research, we provide comprehensive details on our methodology and experimental settings. The datasets used in our experiments are fully described in Section 4 and Appendix A. Our analysis is based on the Neural Regression Collapse (NRC1) metric (Section 3.1) and intrinsic dimension, which is estimated using the 2-NN algorithm detailed in Algorithm 1 (Appendix C). All experimental hyperparameters, including model architectures and training parameters for all tasks, are listed in Appendix A (Tables 3 and 4).

## Ethics Statement

This research focuses on the fundamental geometric properties of neural networks in multivariate regression. The datasets used are standard, publicly available benchmarks in the machine learning and robotics communities and do not contain any personally identifiable or sensitive information. We do not foresee any direct negative societal or ethical implications arising from our work.

## Use of Large Language Models

We utilized large language models (LLMs) during the preparation of this manuscript. The use of these models was strictly limited to improving the grammar, style, and overall clarity of the text. LLMs were not used for generating any of the core ideas, conducting experiments, analyzing results, or drawing scientific conclusions presented in this work.

# REFERENCES

Michele Allegra, Elena Facco, Francesco Denti, Alessandro Laio, and Antonietta Mira. Data segmentation based on the local intrinsic dimension. *Scientific reports*, 10(1):16449, 2020.

Laurent Amsaleg, Oussama Chelly, Teddy Furon, Stéphane Girard, Michael E Houle, Ken-ichi Kawarabayashi, and Michael Nett. Estimating local intrinsic dimensionality. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 29–38, 2015.

George Andriopoulos, Zixuan Dong, Li Guo, Zifan Zhao, and Keith W. Ross. The prevalence of neural collapse in neural multivariate regression. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Alessio Ansuini, Alessandro Laio, Jakob H Macke, and Davide Zoccolan. Intrinsic dimension of data representations in deep neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. In *International conference on machine learning*, pp. 254–263. PMLR, 2018.

Melih Barsbey, Milad Sefidgaran, Murat A Erdogdu, Gael Richard, and Umut Simsekli. Heavy tails in sgd and compressibility of overparametrized neural networks. *Advances in neural information processing systems*, 34:29364–29378, 2021.

Tolga Birdal, Aaron Lou, Leonidas J Guibas, and Umut Simsekli. Intrinsic dimension, persistent homology and generalization in neural networks. *Advances in neural information processing systems*, 34:6776–6789, 2021.

Léonard Blier and Yann Ollivier. The description length of deep learning models. *Advances in Neural Information Processing Systems*, 31, 2018.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym, 2016.

Johan Samir Obando Ceron, Aaron Courville, and Pablo Samuel Castro. In value-based deep reinforcement learning, a pruned network is a good network. In *Forty-first International Conference on Machine Learning*, 2024.

Claudio Ceruti, Simone Bassis, Alessandro Rozza, Gabriele Lombardi, Elena Casiraghi, and Paola Campadelli. Danco: An intrinsic dimensionality estimator exploiting angle and norm concentration. *Pattern recognition*, 47(8):2569–2581, 2014.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

Ciprian A Corneanu, Meysam Madadi, Sergio Escalera, and Aleix M Martinez. What does it mean to learn in deep networks? and, how does one detect adversarial attacks? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4757–4766, 2019.

Ciprian A Corneanu, Sergio Escalera, and Aleix M Martinez. Computing the testing error without a testing set. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2677–2685, 2020.

Misha Denil, Babak Shakibi, Laurent Dinh, Marc'Aurelio Ranzato, and Nando De Freitas. Predicting parameters in deep learning. *Advances in neural information processing systems*, 26, 2013.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Elena Facco, Maria d'Errico, Alex Rodriguez, and Alessandro Laio. Estimating the intrinsic dimension of datasets by a minimal neighborhood information. *Scientific reports*, 7(1):12140, 2017.

Cong Fang, Hangfeng He, Qi Long, and Weijie J Su. Exploring deep neural networks via layer-peeled model: Minority collapse in imbalanced training. *Proceedings of the National Academy of Sciences*, 118(43):e2103091118, 2021.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4RL: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

Tomer Galanti, András György, and Marcus Hutter. On the role of neural collapse in transfer learning. *arXiv preprint arXiv:2112.15121*, 2021.

Quentin Gallouédec, Edward Emanuel Beeching, Clément ROMAC, and Emmanuel Dellandrea. Jack of all trades, master of some, a multi-purpose transformer agent. In *ICML 2024 Workshop: Aligning Reinforcement Learning Experimentalists and Theorists*, 2024.

Peifeng Gao, Qianqian Xu, Yibo Yang, Peisong Wen, Huiyang Shao, Zhiyong Yang, Bernard Ghanem, and Qingming Huang. Towards demystifying the generalization behaviors when neural collapse emerges. *arXiv preprint arXiv:2310.08358*, 2023.

Tianxiang Gao and Vladimir Jojic. Degrees of freedom in deep neural networks. *arXiv preprint arXiv:1603.09260*, 2016.

Li Guo, George Andriopoulos, Zifan Zhao, Shuyang Ling, Zixuan Dong, and Keith Ross. Cross entropy versus label smoothing: A neural collapse perspective. *arXiv preprint arXiv:2402.03979*, 2024.

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

Takuya Hiraoka, Takahisa Imagawa, Taisei Hashimoto, Takashi Onishi, and Yoshimasa Tsuruoka. Dropout q-functions for doubly efficient reinforcement learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=xCVJMsPv3RT.

Wanli Hong and Shuyang Ling. Neural collapse for unconstrained feature model under cross-entropy loss with imbalanced data. *arXiv preprint arXiv:2309.09725*, 2023.

Daniel Hsu, Ziwei Ji, Matus Telgarsky, and Lan Wang. Generalization bounds via distillation. *arXiv preprint arXiv:2104.05641*, 2021.

Like Hui, Mikhail Belkin, and Preetum Nakkiran. Limitations of neural collapse for understanding generalization in deep learning. *arXiv preprint arXiv:2202.08384*, 2022.

Lucas Janson, William Fithian, and Trevor J Hastie. Effective degrees of freedom: a flawed metaphor. *Biometrika*, 102(2):479–485, 2015.

Yann LeCun, John Denker, and Sara Solla. Optimal brain damage. *Advances in neural information processing systems*, 2, 1989.

Elizaveta Levina and Peter Bickel. Maximum likelihood estimation of intrinsic dimension. *Advances in neural information processing systems*, 17, 2004.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. Measuring the intrinsic dimension of objective landscapes. *arXiv preprint arXiv:1804.08838*, 2018.

Pengyu Li, Xiao Li, Yutong Wang, and Qing Qu. Neural collapse in multi-label learning with pick-all-label loss. *arXiv preprint arXiv:2310.15903*, 2023a.

Qiyang Li, Aviral Kumar, Ilya Kostrikov, and Sergey Levine. Efficient deep reinforcement learning requires regulating overfitting. In *The Eleventh International Conference on Learning Representations*, 2023b.

Xiao Li, Sheng Liu, Jinxin Zhou, Xinyu Lu, Carlos Fernandez-Granda, Zhihui Zhu, and Qing Qu. Understanding and improving transfer learning of deep models via neural collapse. *arXiv preprint arXiv:2212.12206*, 2022.

Zhuang Liu, Xuanlin Li, Bingyi Kang, and Trevor Darrell. Regularization matters in policy optimization - an empirical study on continuous control. In *International Conference on Learning Representations*, 2021.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Cong Lu, Philip J. Ball, Tim G. J. Rudner, Jack Parker-Holder, Michael A Osborne, and Yee Whye Teh. Challenges and opportunities in offline reinforcement learning from visual observations. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

Clare Lyle, Zeyu Zheng, Evgenii Nikishin, Bernardo Avila Pires, Razvan Pascanu, and Will Dabney. Understanding plasticity in neural networks. In *International Conference on Machine Learning*, pp. 23190–23211. PMLR, 2023.

Chuang Ma, Tomoyuki Obuchi, and Toshiyuki Tanaka. Neural collapse in cumulative link models for ordinal regression: An analysis with unconstrained feature model. *arXiv preprint arXiv:2506.05801*, 2025.

Xingjun Ma, Bo Li, Yisen Wang, Sarah M Erfani, Sudanthi Wijewickrema, Grant Schoenebeck, Dawn Song, Michael E Houle, and James Bailey. Characterizing adversarial subspaces using local intrinsic dimensionality. *arXiv preprint arXiv:1801.02613*, 2018a.

Xingjun Ma, Yisen Wang, Michael E Houle, Shuo Zhou, Sarah Erfani, Shutao Xia, Sudanthi Wijewickrema, and James Bailey. Dimensionality-driven learning with noisy labels. In *International Conference on Machine Learning*, pp. 3355–3364. PMLR, 2018b.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Dustin G Mixon, Hans Parshall, and Jianzong Pi. Neural collapse with unconstrained features. *arXiv preprint arXiv:2011.11619*, 2020.

Michal Nauman, Michał Bortkiewicz, Piotr Miłoś, Tomasz Trzcinski, Mateusz Ostaszewski, and Marek Cygan. Overestimation, overfitting, and plasticity in actor-critic: the bitter lesson of reinforcement learning. In *Forty-first International Conference on Machine Learning*, 2024.

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

Phillip Pope, Chen Zhu, Ahmed Abdelkader, Micah Goldblum, and Tom Goldstein. The intrinsic dimension of images and its impact on learning. *arXiv preprint arXiv:2104.08894*, 2021.

PyTorch Contributors. Pytorch torch.optim.adamw documentation. `https://docs.pytorch.org/docs/stable/generated/torch.optim.AdamW.html`, 2025.

Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations. In *Proceedings of Robotics: Science and Systems (RSS)*, 2018.

Bastian Rieck, Matteo Togninalli, Christian Bock, Michael Moor, Max Horn, Thomas Gumbsch, and Karsten Borgwardt. Neural persistence: A complexity measure for deep neural networks using algebraic topology. *arXiv preprint arXiv:1812.09764*, 2018.

Umut Simsekli, Ozan Sener, George Deligiannidis, and Murat A Erdogdu. Hausdorff dimension, heavy tails, and generalization in neural networks. *Advances in Neural Information Processing Systems*, 33:5138–5151, 2020.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 2014.

Peter Súkeník, Christoph H Lampert, and Marco Mondelli. Neural collapse is globally optimal in deep regularized resnets and transformers. *arXiv preprint arXiv:2505.15239*, 2025.

Taiji Suzuki, Hiroshi Abe, Tomoya Murata, Shingo Horiuchi, Kotaro Ito, Tokuma Wachi, So Hirai, Masatoshi Yukishima, and Tomoaki Nishimura. Spectral pruning: Compressing deep neural networks via spectral analysis and its generalization error. *arXiv preprint arXiv:1808.08558*, 2018.

Taiji Suzuki, Hiroshi Abe, and Tomoaki Nishimura. Compression based bound for non-compressed network: unified generalization error analysis of large compressible deep neural network. *arXiv preprint arXiv:1909.11274*, 2019.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, Timothy Lillicrap, and Martin Riedmiller. Deepmind control suite, 2018.

Christos Thrampoulidis, Ganesh Ramachandra Kini, Vala Vakilian, and Tina Behnia. Imbalance trouble: Revisiting neural-collapse geometry. *Advances in Neural Information Processing Systems*, 35:27225–27238, 2022.

Tom Tirer and Joan Bruna. Extended unconstrained features model for exploring deep neural collapse. In *International Conference on Machine Learning*, pp. 21478–21505. PMLR, 2022.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, 2012. doi: 10.1109/IROS.2012.6386109.

Mark Towers, Jordan K. Terry, Ariel Kwiatkowski, John U. Balis, Gianluca de Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Arjun KG, Markus Krimmel, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Andrew Tan Jin Shen, and Omar G. Younis. Gymnasium, March 2023. URL `https://zenodo.org/record/8127025`.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Markus Krimmel, Arjun KG, et al. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*, 2024.

Siwei Wang and Stephanie E Palmer. Towards understanding neural collapse in supervised contrastive learning with the information bottleneck method. *arXiv preprint arXiv:2305.11957*, 2023.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45. Association for Computational Linguistics, 2020.

Robert Wu and Vardan Papyan. Linguistic collapse: Neural collapse in (large) language models. *Advances in Neural Information Processing Systems*, 37:137432–137473, 2024.

Yibo Yang, Shixiang Chen, Xiangtai Li, Liang Xie, Zhouchen Lin, and Dacheng Tao. Inducing neural collapse in imbalanced learning: Do we really need a learnable classifier at the end of deep neural network? *Advances in neural information processing systems*, 35:37991–38002, 2022.

Can Yaras, Peng Wang, Zhihui Zhu, Laura Balzano, and Qing Qu. Neural collapse with normalized features: A geometric analysis over the riemannian manifold. *Advances in neural information processing systems*, 35:11547–11560, 2022.

Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Mastering visual continuous control: Improved data-augmented reinforcement learning. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=_SJ-_yyes8`.

Fan Yin, Jayanth Srinivasa, and Kai-Wei Chang. Characterizing truthfulness in large language model generations with local intrinsic dimension. *arXiv preprint arXiv:2402.18048*, 2024.

Omar G. Younis, Rodrigo Perez-Vicente, John U. Balis, Will Dudley, Alex Davey, and Jordan K Terry. Minari, September 2024. URL `https://doi.org/10.5281/zenodo.13767625`.

Jinxin Zhou, Xiao Li, Tianyu Ding, Chong You, Qing Qu, and Zhihui Zhu. On the optimization landscape of neural collapse under MSE loss: Global optimality with unconstrained features. In *International Conference on Machine Learning*, pp. 27179–27202. PMLR, 2022a.

Jinxin Zhou, Chong You, Xiao Li, Kangning Liu, Sheng Liu, Qing Qu, and Zhihui Zhu. Are all losses created equal: A neural collapse perspective. *Advances in Neural Information Processing Systems*, 35:31697–31710, 2022b.

Wei Zhu, Qiang Qiu, Jiaji Huang, Robert Calderbank, Guillermo Sapiro, and Ingrid Daubechies. Ldmnet: Low dimensional manifold regularized neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2743–2751, 2018.

Zhihui Zhu, Tianyu Ding, Jinxin Zhou, Xiao Li, Chong You, Jeremias Sulam, and Qing Qu. A geometric analysis of neural collapse with unconstrained features. *Advances in Neural Information Processing Systems*, 34:29820–29834, 2021.

# A Experiment Details

## A.1 MuJoCo experiments

MuJoCo (Multi-Joint dynamics with Contact) is a physics engine designed for research in robotics, biomechanics, and animation, providing fast and accurate simulations of systems involving complex contact dynamics. It balances physical realism with computational efficiency to enable reliable modeling of robot–environment interactions (Towers et al., 2024). Environments involved in this work include:

- **Reacher**: A two-jointed robotic arm tasked with moving its tip to a randomly generated target in a 2D plane.
- **Swimmer**: A chain-like robot with three body segments connected by two rotors, aiming to propel itself forward in 2D as quickly as possible.
- **Hopper**: A one-legged, four-part robot that seeks to hop forward at maximum speed in 2D.
- **HalfCheetah**: A planar, bipedal robot with a torso and two legs, each consisting of two joints. It aims to run forward as quickly as possible along a 2D track by coordinating its leg movements.
- **Ant**: A quadrupedal robot with four legs and multiple joints, designed to move in a 3D plane. Its goal is to walk or run forward efficiently, despite the challenge of balancing and coordinating many degrees of freedom. Although Ant's state space consists of 111 dimensions, 84 of the dimensions related to external contact forces are always zeros in the dataset. Thus, the effective input dimension is 27.

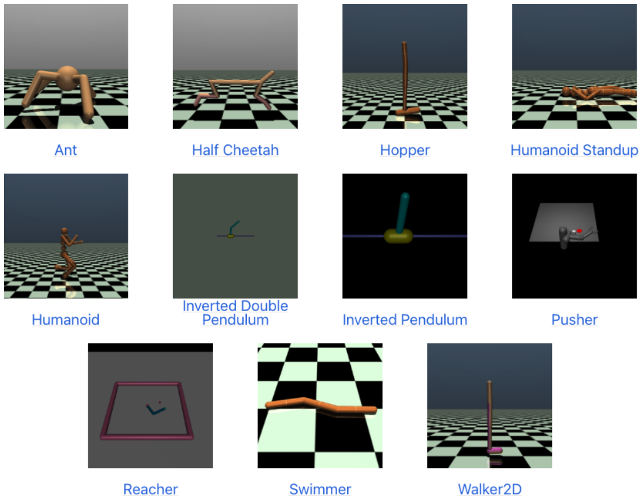


Figure 7: Screenshot of various MuJoCo environments (Towers et al., 2024).

All environments introduce stochasticity by perturbing a fixed initial state with Gaussian noise. Their state spaces combine positions of body and joint with corresponding velocities. Control is achieved by applying joint torques, which serve as the actions. Expert datasets are generated by first training policies through online reinforcement learning (Fu et al., 2020; Gallouédec et al., 2024) until high performance, then executing these policies to produce trajectories of states $\mathbf{x}_i$ and actions $\mathbf{y}_i$. Here, $\mathbf{x}_i$ encodes robot positions, joint angles, velocities, and angular velocities, while $\mathbf{y}_i$ denotes the applied joint torques.

An episode of expert demonstration has a length of 50 for Reacher, and it has a length of 1,000 for all other environments. Thus, by taking 20,000 data points from each expert dataset, the regression model learns from at least 20 complete trajectories to clone the expert's behavior. For evaluation, we retain a subset of the full validation dataset, keeping the number of data points at 20% of the training data size. For small datasets (1K) used in Figures 6 and 10, the test datasets contain 1,000 unseen samples.

There is no absolute threshold for what constitutes "low-data" versus "high-data," as this depends on problem complexity and data structure. In our experiments, we operationally define this distinction using datasets of 1,000 samples versus 20,000 samples—a 20-fold difference that produces qualitatively different generalization behavior. In the low-data regime, the training set provides sparse coverage of the true data manifold, introducing sampling artifacts such as spurious correlations and outlier effects that do not reflect the true underlying distribution. Models trained on such small datasets, particularly without sufficient regularization, tend to memorize these sample-specific patterns rather than learning generalizable structure. In the high-data regime with 20,000 samples, the training set provides denser coverage, the effect of sampling artifacts diminishes, and the empirical distribution more closely approximates the true underlying distribution.

Table 3 summarizes all model hyperparameters and experimental settings for MuJoCo datasets. A subset of possible hyperparameter combinations is used for each figure:

- Figure 1 plots the min-max normalized Test MSE as a function of the min-max normalized NRC1 values for the model architecture 3-256 (3 hidden layers and 256 hidden units) and all possible weight decay values.
- Figure 3 and 8 record NRC1 values along the training process. The model architecture is fixed to 3-256 for all datasets. And we show 10 weight decay values in $\{0, 0.0001, 0.0003, 0.0005, 0.0007, 0.001, 0.003, 0.005, 0.007, 0.01\}$.
- Figure 4 establishes the relationship between NRC1 and $ID_H$. Each subplot includes all weight decays listed in Table 3. And each weight decay is combined with 9 model architectures in {3-64, 3-128, 3-256, 3-512, 3-1024, 1-256, 2-256, 4-256, 5-256}.
- Figure 5 and 9 depict how intrinsic dimension evolves for each network layer. The model architecture is fixed at 3-256 and the title of each subplot annotates the weight decay value.
- Figure 6 and 10 empirically reveal how generalization ability is affected by $ID_H$. We focus on a single model depth of 3, and vary the model width among $\{64, 128, 256, 512, 1024\}$. For each model architecture, we evaluate all possible weight decay values listed in Table 3.
- Figure 11 and 12 follow the same experimental setup as above (Figs. 6 and 10), but emphasize on the comparison between $ID_H$ and $ID_P$.

Table 3: All hyperparameter settings involved for experiments on MuJoCo datasets. Each figure employs a subset of possible hyperparameter combinations.

| | **Hyperparameter** | **Value** |
|---|---|---|
| Model Architecture | Number of hidden layers | $\{1, 2, 3, 4, 5\}$ |
| | Hidden layer dimension | $\{64, 128, 256, 512, 1024\}$ |
| | Activation function | ReLU |
| | Number of linear projection layer (**W**) | 1 |
| Training | Epochs | $3 \times 10^5$ (20K-datasets) |
| | | $5 \times 10^6$ (1K-datasets) |
| | Batch size | 4096 (20K-datasets) |
| | | 1000 (1K-datasets) |
| | Optimizer | SGD |
| | Learning rate | $1 \times 10{-}2$ |
| | Weight decay | $\{0, 1e^{-5}, 1e^{-4}, 3e^{-4}, 5e^{-4}, 7e^{-4}, 1e^{-3}, 3e^{-3}\}$, Reacher |
| | | $\{0, 1e^{-5}, 1/3/5/7e^{-4}, 1/3/5/7e^{-3}, 1e^{-2}, 3e^{-2}\}$, Otherwise |
| | Seed | 0 |
| | Compute resources | NVIDIA A100 40GB |
| | Number of CPU compute workers | 4 |
| | Requested compute memory | 16 GB |
| | Average training time per model | 20 hours |

### A.2 MNIST/CIFAR10 EXPERIMENTS

The regression models for both the MNIST and CIFAR-10 tasks were trained across a spectrum of hyperparameters to thoroughly investigate the effects of architecture and regularization on the learned representations. The specific settings for model architecture, optimizer, and other training parameters are detailed in Table 4.

Table 4: All hyperparameter settings involved for experiments on MNIST and CIFAR-10 datasets.

| | **Hyperparameter** | **Value** |
|---|---|---|
| Model Architecture | Number of hidden layers | 3 |
| | Hidden layer dimension | $\{32, 64, 128, 256, 512\}$ |
| | Activation function | ReLU |
| Training | Epochs | 200 |
| | Batch size | 64 |
| | Optimizer | Adam |
| | Learning rate | $\{1 \times 10^{-3}, 5 \times 10^{-3}\}$ |
| | Weight decay (MNIST) | $\{0, 10^{-5}, 10^{-4}, 3 \times 10^{-4}, 5 \times 10^{-4}, 7 \times 10^{-4}, 10^{-3}, 3 \times 10^{-3}, 7 \times 10^{-3}\}$ |
| | Weight decay (CIFAR-10) | $\{0, 10^{-5}, 10^{-4}, 3 \times 10^{-4}, 5 \times 10^{-4}, 7 \times 10^{-4}, 10^{-3}\}$ |
| | Seed | 0 |
| | Compute resources | NVIDIA A100 80GB |
| | Average training time per model | 2 hours |

We define noisy-target tasks as tasks where the regression targets contain information that causes models to learn patterns that do not generalize to unseen data. The term "noise" here does not refer to random measurement error or label corruption, but rather to information that varies between semantically similar examples due to instance-specific characteristics rather than reflecting the true underlying function.

The synthetic MNIST regression task represents a low-noise setting. The feature extractor is a CNN trained specifically on MNIST until achieving over 99% classification accuracy, creating a self-consistent target-generation pipeline. During training, the CNN learns to discard instance-specific variations (such as stroke thickness, slight rotations, or pixel-level noise) that are irrelevant for predicting the target label, retaining only the semantically meaningful information. As a result, the mapping from MNIST images to projected features is smooth and well-aligned with the data domain, enabling models to generalize effectively (Figs. 6 (p),(q),(r)).

In contrast, the synthetic CIFAR-10 regression task is a noisy-target setting due to domain mismatch. CIFAR-10 images are processed through a ResNet-18 pretrained on ImageNet to extract features, which are then projected to 10-dimensional targets. The ImageNet encoder captures fine-grained visual details—texture patterns, color distributions, edge structures—optimized for ImageNet's 1,000 classes. When applied to CIFAR-10's 32×32 images, these features encode not only semantic content but also instance-specific characteristics: two images of cars may differ in color, lighting, rendering artifacts, or background elements, all of which significantly influence the extracted features. Models can fit these instance-specific components during training, achieving low training error. However, at test time, new images from the same semantic classes exhibit different instance-specific details. The learned mappings for these non-generalizable components do not transfer, resulting in poor generalization. This is evident in Figs. 6 (g),(h),(i), where models achieve low training MSE but high test MSE, characteristic of overfitting to target noise.

# B Additional Experiments

This section lists additional results that complement the experiments in the main body for all considered datasets. Figure 3 and 8 depict NRC1 evolution along the training process. Then, Figure 5 and 9 record ID evolution along the training process. Finally, Figure 6 and 10 shows how generalization power correlates with $ID_H$.

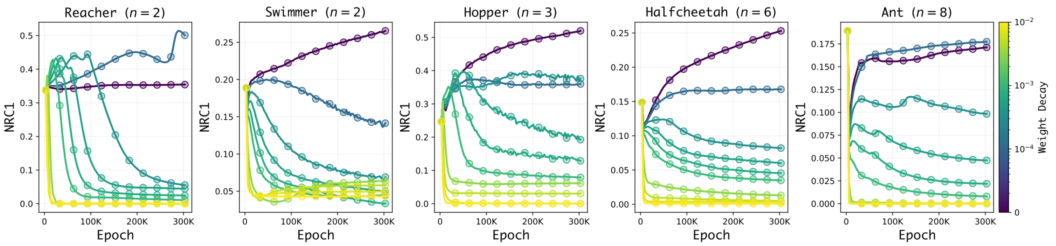


Figure 8: NRC1 decreases as weight decay becomes stronger, leading to model collapse.

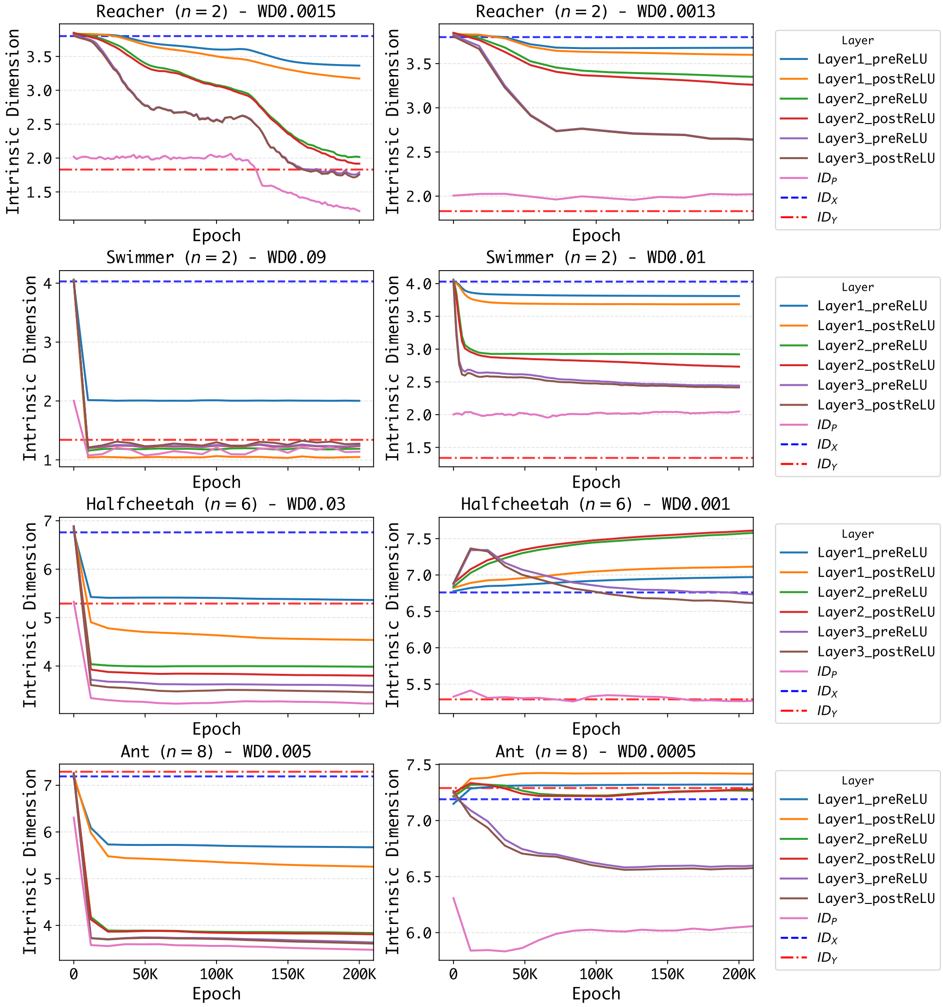


Figure 9: Intrinsic dimension of input, output, and hidden layers over training epochs for a collapsed (left) and a non-collapsed model (right) for the Reacher dataset. Each subfigure shows the evolution of intrinsic dimension across layers with blue, orange dashed and pink lines denoting the intrinsic dimension of inputs, targets, and predicted outputs, respectively.

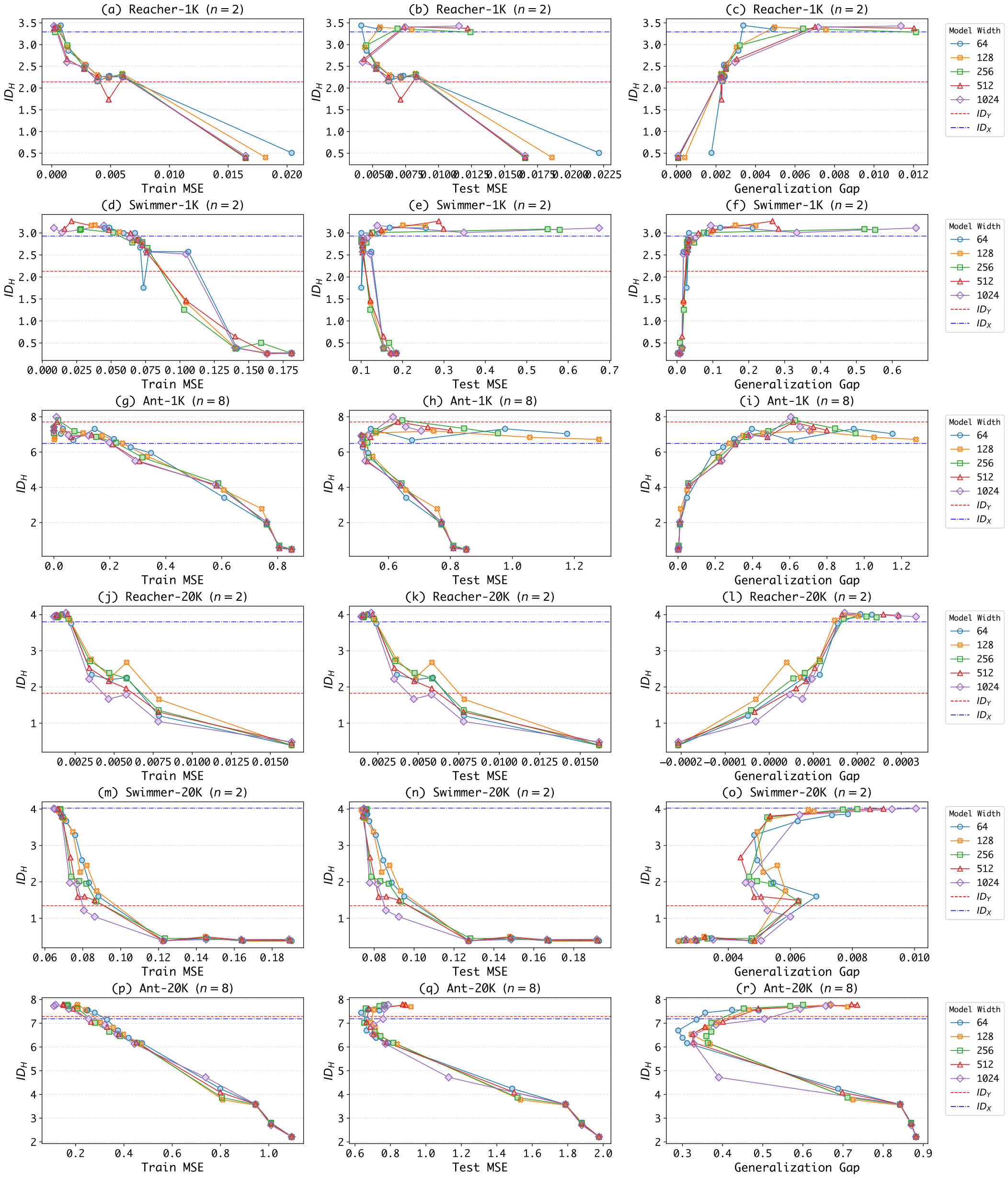


Figure 10: Generalization ability and Intrinsic Dimension for the MuJoCo datasets

## C Intrinsic Dimension and the 2-NN Algorithm

The intrinsic dimension (ID) of a dataset is the minimum number of coordinates needed to represent the data faithfully. If data points lie on or near a $d$-dimensional manifold $\mathcal{M}$ embedded in $\mathbb{R}^D$ with $d \ll D$, then $d$ is the intrinsic dimension. For example, a circle in 3D space has $d = 1$ and a sphere surface in 10D has $d = 2$.

A critical distinction exists between PCA dimensionality and intrinsic dimension. Consider a 1D spiral embedded in $\mathbb{R}^{10}$ parameterized by $t$. The spiral winds through space with substantial variance across all 10 axes, requiring multiple PCA components to capture the signal. Yet, the intrinsic dimension is exactly $d = 1$, because specifying a single scalar value, such as the arc length from the origin, is sufficient to uniquely locate any point on the curve. This illustrates that curved or folded manifolds can require many linear directions to approximate while having low intrinsic dimension. Figure 2 demonstrates this for neural regression: collapsed features lie near a 2D linear subspace (yellow plane) yet occupy a nonlinear 1D manifold within it.

Intrinsic dimension has emerged as a powerful tool for quantifying representational capacity in neural networks. Ansuini et al. (2019) showed that ID evolves systematically across CNN layers, with models achieving lower ID in penultimate layers exhibiting superior generalization. Li et al. (2018) used ID to measure effective dimensionality of parameter spaces, finding that models traversing lower-dimensional solution manifolds generalize better. Pope et al. (2021) demonstrated that ID of image datasets directly impacts sample complexity, while Yin et al. (2024) recently used per-sample ID to identify untruthful LLM outputs.

We estimate ID using the 2-NN estimator (Facco et al., 2017), which exploits a fundamental geometric property: in a $d$-dimensional space, the probability of finding neighbors within a given distance scales with dimension $d$. The key insight is to consider not absolute distances, but the ratio $\mu = r_2/r_1 \geq 1$ of the second to first nearest-neighbor distances. Remarkably, under the assumption of locally uniform density (density approximately constant within the range of the second neighbor), this ratio has a distribution that depends only on the intrinsic dimension $d$, with the local density completely canceling out. Specifically, the cumulative distribution function of $\mu$ is

$$F(\mu) = 1 - \mu^{-d}, \quad \mu \geq 1$$

This property makes the estimator robust to density variations, since we never need to estimate density itself. Taking logarithms yields the linear relationship

$$\log(1 - F(\mu)) = -d \log \mu$$

The 2-NN algorithm estimates $d$ by computing $\mu_i$ for each point, constructing the empirical CDF $F_{\text{emp}}$, and performing linear regression on the transformed coordinates $\{(\log \mu_i, -\log(1 - F_{\text{emp}}(\mu_i)))\}$. The requirement of local uniformity only within the second-neighbor distance is much weaker than global uniformity, making the estimator practical for real datasets with varying density and curvature.

**Algorithm 1:** 2-NN Intrinsic Dimension Estimation

```
Input: Dataset $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^{M}$.
Output: Estimated intrinsic dimension $\hat{d}$.
for $i \leftarrow 1$ to $M$ do
    Compute Euclidean distances to the first and second nearest neighbors, $r_1(\mathbf{x}_i)$ and $r_2(\mathbf{x}_i)$;
    Compute ratio $\mu_i \leftarrow r_2(\mathbf{x}_i)/r_1(\mathbf{x}_i)$;
Sort the ratios such that $\mu_{\sigma(1)} \leq \mu_{\sigma(2)} \leq \cdots \leq \mu_{\sigma(M)}$;
for $i \leftarrow 1$ to $M$ do
    Assign empirical CDF value $F_{emp}(\mu_{\sigma(i)}) \leftarrow \frac{i}{M}$;
Construct the coordinate set for regression:
    $$\mathcal{S} \leftarrow \Big\{ \big( \log \mu_{\sigma(i)},\ -\log\big(1 - F_{emp}(\mu_{\sigma(i)})\big) \big) \Big\}_{i=1}^{M-1}$$
Fit a line through the origin to $\mathcal{S}$ using least squares;
return Slope of the fitted line ($\hat{d}$);
```

## D PROOF OF THEOREM 1

The proof of Theorem 1 follows directly from Sard's theorem.

*Proof.* Let $g : \mathcal{M} \to \mathcal{N}$ be a smooth map where $\dim(\mathcal{M}) = m$ and $\dim(\mathcal{N}) = n$, under the condition $m < n$. Consider an arbitrary point $p \in \mathcal{M}$. The differential of the map at this point, $dg_p : T_p\mathcal{M} \to T_{g(p)}\mathcal{N}$, is a linear transformation from the $m$-dimensional tangent space of $\mathcal{M}$ at $p$ to the $n$-dimensional tangent space of $\mathcal{N}$ at $g(p)$.

By the rank-nullity theorem, the rank of $dg_p$ is bounded by the dimension of its domain, so it holds that rank$(dg_p) \leq m$. Given that $m < n$, it follows that rank$(dg_p) < n$. A linear map is surjective if and only if its rank equals the dimension of its codomain; thus, $dg_p$ is not surjective.

As the choice of $p$ was arbitrary, this holds for all $p \in \mathcal{M}$. By definition, a point is critical if its differential is not surjective. Therefore, every point in the domain $\mathcal{M}$ is a critical point of $g$. The image of the set of critical points is the set of critical values. In this case, the set of critical values is the entire image of the map, $g(\mathcal{M})$.

By Sard's Theorem, the set of critical values of a smooth map has Lebesgue measure zero in the codomain. It follows that the image $g(\mathcal{M})$ has measure zero in $\mathcal{N}$. However, a smooth $n$-dimensional manifold (for $n \geq 1$) has positive Lebesgue measure. Since a set of measure zero cannot be equal to a set of positive measure, it must be that $g(\mathcal{M}) \neq \mathcal{N}$.

Therefore, the map $g$ is not surjective. □

## E INTRINSIC DIMENSION AND THE OUTPUT LAYER

We consider here the intrinsic dimension of the outputs (equivalently, the final predictions), $ID_P$. We will see that here too the relationship between intrinsic dimension and generalization exhibits key differences between classification and regression.

With respect to the output layer, a structural constraint arises from the classification setting. Specifically, the intrinsic dimension of the output layer necessarily satisfies

$$\log_2 C \leq ID_P \leq C,$$

where $C$ is the number of classes. Empirical results consistently show $ID_P$ equals the lower bound of this inequality if the model generalizes well. We refer the reader to the discussion in Section 3.1 of (Ansuini et al., 2019). Conversely, saturation of the upper bound, i.e., $ID_P \simeq C$, is associated with poor generalization performance, suggesting that maximal output layer dimensionality corresponds to overfitting in classification tasks, see Section 3.5 in Ansuini et al. (2019).

In contrast, for neural multivariate regression, the structure of the output leads to the trivial bound

$$1 \leq ID_P \leq n,$$

where $n$ is the number of output variates. Interestingly, our empirical findings reveal a departure from the classification setting. As shown in the middle column in Figures 11-12, when test MSE is low, the intrinsic dimension of the output layer, $ID_P$ satisfies $ID_P \simeq ID_Y$, which can be close to $n$, saturating the upper bound of the inequality above. Notably, unlike in classification, this saturation is associated with improved test performance. By contrast, when $ID_P$ falls below $ID_Y$, test MSE performance deteriorates, see Figures 11-12.

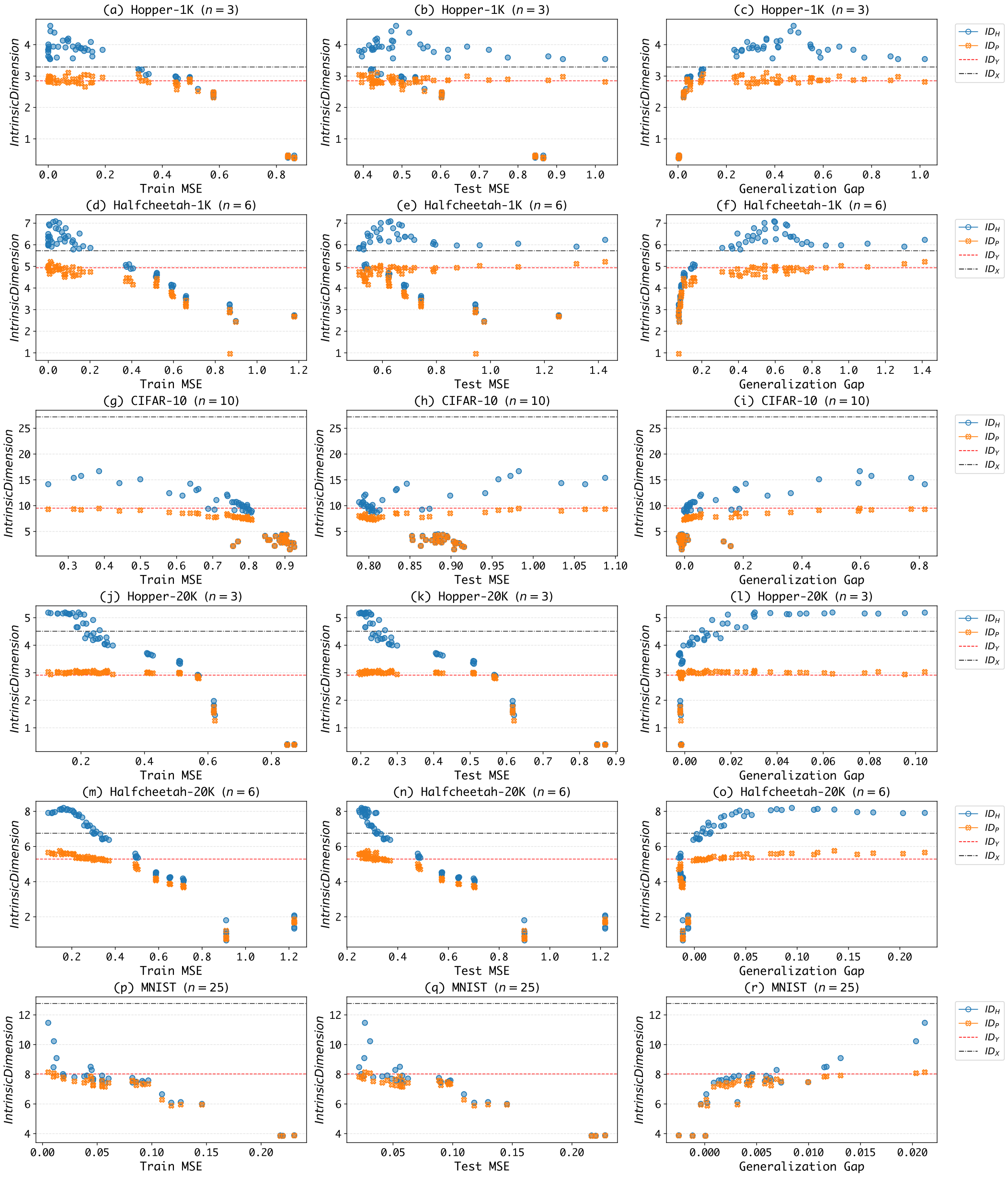


Figure 11: Comparison between $ID_H$ and $ID_P$ for Halfcheetah, Hopper, CIFAR-10, and MNIST datasets

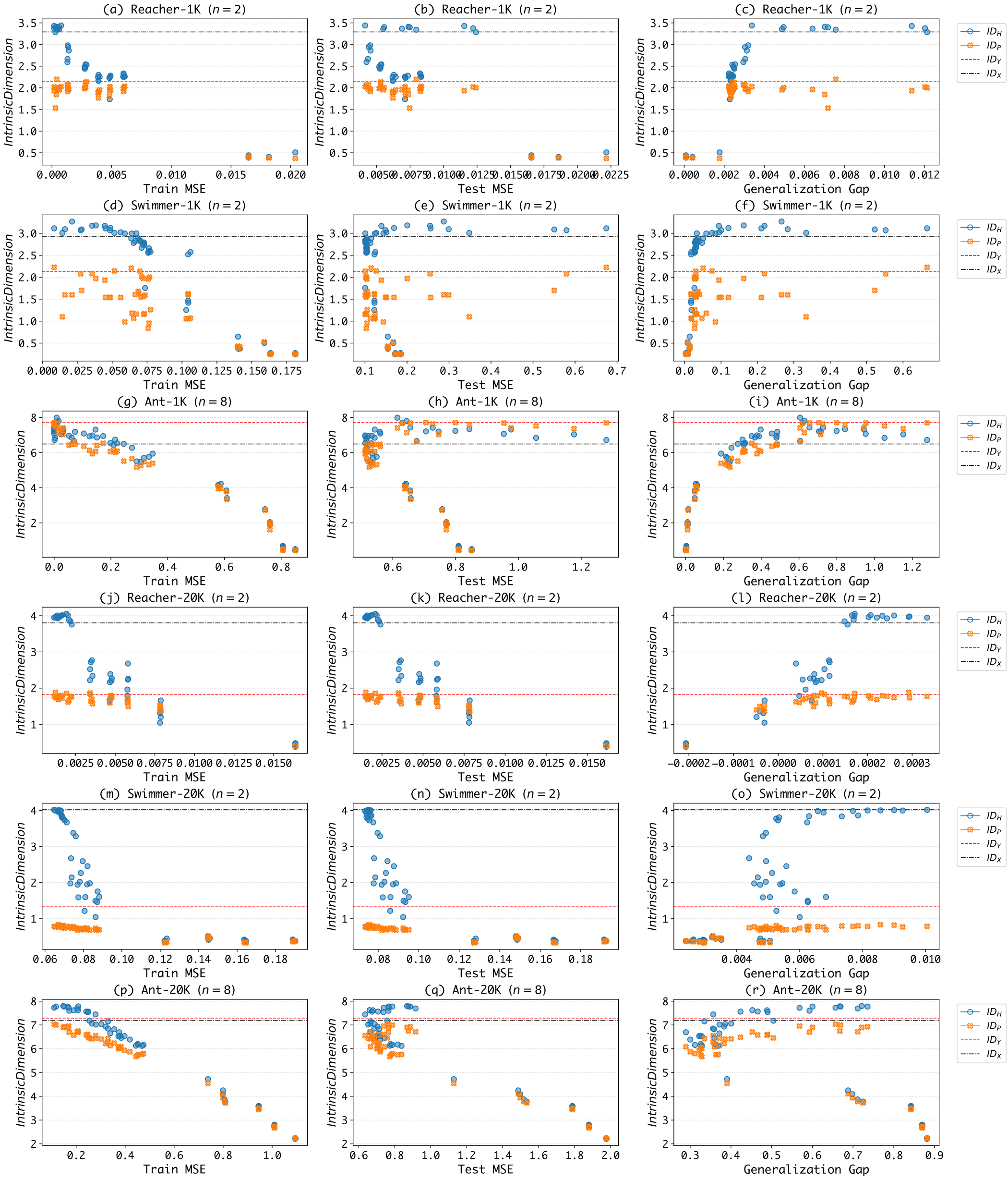


Figure 12: Comparison between $ID_H$ and $ID_P$ for Reacher, Swimmer and Ant datasets

# F Why Weight Decay Leads to Collapse: Analysis via the Unconstrained Feature Model

In this section, we provide a theoretical explanation for why weight decay causes neural regression collapse through the lens of the Unconstrained Feature Model (UFM). The UFM is a simplified mathematical abstraction that does not capture all aspects of practical neural networks, but provides insight into the geometric mechanisms by which regularization constrains learned representations.

The UFM abstracts the neural regression problem by treating the feature extractor $h_\theta(\cdot)$ as producing arbitrary feature vectors $\mathbf{h}_i \in \mathbb{R}^d$ for each input $\mathbf{x}_i$, collected into a feature matrix $H \in \mathbb{R}^{d\times M}$. The final prediction is obtained via a linear map $W \in \mathbb{R}^{n\times d}$ and bias $\mathbf{b} \in \mathbb{R}^n$, giving predictions $\hat{Y} = WH + \mathbf{b}\mathbf{1}^\top$. The UFM objective is:

$$\min_{W,H,\mathbf{b}} \frac{1}{2M}\|WH + \mathbf{b}\mathbf{1}^\top - Y\|_F^2 + \frac{\lambda_W}{2}\|W\|_F^2 + \frac{\lambda_H}{2}\|H\|_F^2 \tag{1}$$

The model is "unconstrained" because $H$ can be any matrix in $\mathbb{R}^{d\times M}$, unlike in actual neural networks where $H$ is constrained by the input data and network architecture. In the UFM we typically have $d \gg n$, mirroring the overparameterized regime where feature dimension greatly exceeds target dimension.

The optimal bias is $\mathbf{b}^* = \bar{\mathbf{y}}$ where $\bar{\mathbf{y}} = \frac{1}{M}\sum_{i=1}^M \mathbf{y}_i$. Defining the centered target matrix $\tilde{Y} = Y - \bar{\mathbf{y}}\mathbf{1}^\top \in \mathbb{R}^{n\times M}$, the problem reduces to:

$$\min_{W,H} \frac{1}{2M}\|WH - \tilde{Y}\|_F^2 + \frac{\lambda_W}{2}\|W\|_F^2 + \frac{\lambda_H}{2}\|H\|_F^2 \tag{2}$$

We denote the empirical covariance by $\Sigma = \frac{1}{M}\tilde{Y}\tilde{Y}^\top \in \mathbb{R}^{n\times n}$, and assume $\Sigma$ is full rank with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n > 0$. A key quantity is $c = \lambda_W\lambda_H$.

We first restate the characterization of global minimizers when regularization is present, adapted from Theorem 4.1 of Andriopoulos et al. (2024). We focus on the regime where $c < \lambda_n$ since we are interested in the limiting behavior when $\lambda_H \to 0$ and $\lambda_W \to 0$.

**Theorem 2** (Regularized UFM Solution, adapted from Andriopoulos et al. (2024))**.** *Suppose $0 < c < \lambda_n$. Define $A = \Sigma^{1/2} - \sqrt{c}I_n$. Then any global minimizer $(W_\lambda, H_\lambda)$ of equation 2 can be expressed as:*

$$W_\lambda = \left(\frac{\lambda_H}{\lambda_W}\right)^{1/4} A^{1/2}R \tag{3}$$

$$H_\lambda = \left(\frac{\lambda_W}{\lambda_H}\right)^{1/4} R^\top A^{1/2}(\Sigma^{1/2})^{-1}\tilde{Y} \tag{4}$$

*where $R \in \mathbb{R}^{n\times d}$ is any matrix satisfying $RR^\top = I_n$.*

*Proof.* This follows from Theorem 4.1 of Andriopoulos et al. (2024) with $c < \lambda_n$ implying $j^* = n$. □

The regularized solution has a specific dimensional structure. Since $A^{1/2}(\Sigma^{1/2})^{-1}\tilde{Y} \in \mathbb{R}^{n\times M}$ and $R^\top \in \mathbb{R}^{d\times n}$, the matrix $H_\lambda \in \mathbb{R}^{d\times M}$ has columns lying in the column space of $R^\top$, which is at most $n$-dimensional. Even though $H_\lambda$ lives in a $d$-dimensional ambient space with $d \gg n$, its columns are confined to an $n$-dimensional subspace.

When regularization is absent, the problem has infinitely many global minimizers, characterized by the following theorem from Andriopoulos et al. (2024).

**Theorem 3** (Unregularized UFM Solutions, from Andriopoulos et al. (2024))**.** *When $\lambda_W = \lambda_H = 0$, a pair $(W, H)$ is a global minimizer of equation 2 if and only if $WH = \tilde{Y}$ and $W$ has full row rank. For any such $W$, the corresponding global minimizers in $H$ are:*

$$H_{unreg} = W^\dagger\tilde{Y} + (I_d - W^\dagger W)Z \tag{5}$$

*where $W^\dagger$ is the Moore–Penrose pseudoinverse and $Z \in \mathbb{R}^{d\times M}$ is arbitrary.*

*Proof.* See Theorem 4.3 of Andriopoulos et al. (2024). □

The structure in equation 5 decomposes solutions into two orthogonal components: $W^\dagger \tilde{Y}$ lies in the row space of $W$ (dimension at most $n$), while $(I_d - W^\dagger W)Z$ lies in the null space of $W$ (dimension exactly $d - n$). The Frobenius norm satisfies $\|H_{\text{unreg}}\|_F^2 = \|W^\dagger \tilde{Y}\|_F^2 + \|(I_d - W^\dagger W)Z\|_F^2$ by orthogonality. The minimum-norm solution is achieved when $Z = 0$, eliminating the $(d - n)$-dimensional null space component. When $Z \neq 0$, the null space allows $H$ to span up to $d$ dimensions, whereas $Z = 0$ confines $H$ to at most $n$ dimensions.

We now investigate what happens to the regularized solution as weight decay vanishes, examining the limit $\lambda_W, \lambda_H \to 0$ while maintaining $\lambda_H/\lambda_W = k > 0$.

**Lemma 1** (Limiting Reconstruction)**.** *Suppose* $\lim_{\lambda_H \to 0, \lambda_W \to 0}(\lambda_H/\lambda_W) = k > 0$ *and let* $(W_\lambda, H_\lambda)$ *be as in Theorem 2. Then:*

$$\lim_{\lambda_W, \lambda_H \to 0} W_\lambda H_\lambda = \tilde{Y} \tag{6}$$

*Proof.* Since $c = \lambda_W \lambda_H \to 0$, Theorem 2 applies for sufficiently small $\lambda_W, \lambda_H$. Multiplying:

$$\begin{aligned} W_\lambda H_\lambda &= \left(\frac{\lambda_H}{\lambda_W}\right)^{1/4} A^{1/2} R \cdot \left(\frac{\lambda_W}{\lambda_H}\right)^{1/4} R^\top A^{1/2} (\Sigma^{1/2})^{-1} \tilde{Y} \qquad (7) \\ &= A^{1/2}(RR^\top)A^{1/2}(\Sigma^{1/2})^{-1}\tilde{Y} = A(\Sigma^{1/2})^{-1}\tilde{Y} \qquad (8) \end{aligned}$$

Substituting $A = \Sigma^{1/2} - \sqrt{c} I_n$:

$$W_\lambda H_\lambda = (\Sigma^{1/2} - \sqrt{c} I_n)(\Sigma^{1/2})^{-1}\tilde{Y} = (I_n - \sqrt{c}\Sigma^{-1/2})\tilde{Y} \tag{9}$$

As $c \to 0$, this yields $\tilde{Y}$. □

**Theorem 4** (Limiting Solution Structure)**.** *Under the assumptions of Lemma 1, define:*

$$\begin{aligned} W_0 &= \lim_{\lambda_W, \lambda_H \to 0} W_\lambda = k^{1/4}\Sigma^{1/4} R \qquad (10) \\ H_0 &= \lim_{\lambda_W, \lambda_H \to 0} H_\lambda = k^{-1/4} R^\top \Sigma^{-1/4} \tilde{Y} \qquad (11) \end{aligned}$$

*Then* $(W_0, H_0)$ *is a global minimizer of the unregularized problem with:*

$$H_0 = W_0^\dagger \tilde{Y} \tag{12}$$

*In particular,* $H_0$ *has no null space component (i.e.,* $Z = 0$ *in Theorem 3).*

*Proof.* From Lemma 1, $W_0 H_0 = \tilde{Y}$, so $(W_0, H_0)$ is a global minimizer. Since $W_0 = k^{1/4}\Sigma^{1/4}R$ has full row rank, its pseudoinverse is:

$$W_0^\dagger = W_0^\top (W_0 W_0^\top)^{-1} = (k^{1/4}\Sigma^{1/4}R)^\top (k^{1/2}\Sigma^{1/2})^{-1} = k^{-1/4}R^\top \Sigma^{-1/4} \tag{13}$$

Thus $W_0^\dagger \tilde{Y} = k^{-1/4} R^\top \Sigma^{-1/4}\tilde{Y} = H_0$, confirming $Z = 0$. For any $H$ satisfying $W_0 H = \tilde{Y}$, we have $H = W_0^\dagger \tilde{Y} + (I_d - W_0^\dagger W_0)Z$ with:

$$\|H\|_F^2 = \|W_0^\dagger \tilde{Y}\|_F^2 + \|(I_d - W_0^\dagger W_0)Z\|_F^2 \geq \|H_0\|_F^2 \tag{14}$$

by orthogonality, with equality if and only if $Z = 0$. Thus $H_0$ is the minimum-norm solution. □

These results explain why weight decay causes dimensional collapse. Without regularization, $H$ can utilize the full $d$-dimensional space through the arbitrary null space component $(I_d - W^\dagger W)Z$, where the null space has dimension $d - n$. With any positive regularization, Theorem 2 shows $H_\lambda$ is confined to an at most $n$-dimensional subspace. Theorem 4 proves that even as $\lambda_W, \lambda_H \to 0^+$, the limiting solution has $Z = 0$, eliminating the $(d - n)$-dimensional null space component. This demonstrates that even infinitesimally small weight decay induces dimensional collapse by biasing toward the minimum-norm solution of the unregularized problem, which lies entirely in the $n$-dimensional row space of $W$. Since typically $d \gg n$, this represents a massive dimensional reduction from the ambient feature space to a low-dimensional subspace determined by the target structure.

# G Neural Regression Collapse without Weight Decay

Weight decay is a canonical and widely adopted model regularization technique for preventing large models from overfitting to data. Real-world applications include but are not limited to (1) regularizing transformer backbones for large language models (Wolf et al., 2020) and robotic generalist policies (Chen et al., 2021); (2) participating, by default, in the common Pytorch implementation of AdamW optimizer (Loshchilov & Hutter, 2019; PyTorch Contributors, 2025) with `weight_decay=0.01`; (3) improving sample efficiency of online reinforcement learning algorithms (Liu et al., 2021; Li et al., 2023b); (4) and facilitating research in model plasticity in deep learning (Lyle et al., 2023; Nauman et al., 2024; Ceron et al., 2024). In this section, we examine other factors that can affect neural regression collapse under minor control of, or in the absence of, weight decay.

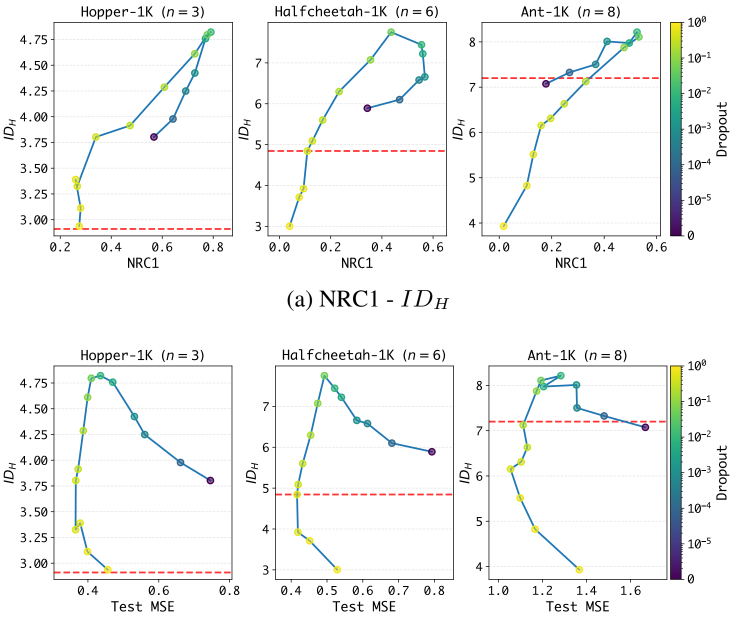
## G.1 Dropout Regularization

Modern implementations, across deep learning domains, continue to rely on dropout regularization (Srivastava et al., 2014) to mitigate overfitting, underscoring its persistent role in practical training, such as computer vision (Dosovitskiy et al., 2021), NLP (Devlin et al., 2019; Wolf et al., 2020), and reinforcement learning (Hiraoka et al., 2022). We empirically analyze how dropout regularization influences neural regression collapse by varying its strength from a large range ($\in \{0, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$). *No* weight decay is applied. In this section, datasets include Hopper, Halfcheetah, and Ant with two sizes, 1K (Figure 13) and 20K (Figure 14). The horizontal red dashed line represents $ID_Y$.

Figure 13(a) and Figure 14(a) show the relationship between $ID_H$ and NRC1. We first confirm the same conclusion as made in Section 5, despite the new regularization. $ID_H$ provides a more refined geometric structure than NRC1. Collapsed models with near-zero NRC1 values have varying $ID_H$ below or in the vicinity of $ID_Y < n$, while non-collapsed models with non-trivial NRC1 maintain their $ID_H$ to be above $ID_Y$ and to be positively correlated with $NRC1$. Interestingly, dropout regularization differs from weight decay in that mild dropout (e.g., $\leq 0.01$) can effectively prevent models from collapse by increasing both NRC1 and $ID_H$. This observation sheds light on a geometric interpretation of the effectiveness of mild dropout in reinforcement learning as proposed by Hiraoka et al. (2022).

Figure 13(b) and Figure 14(b) show the relationship between $ID_H$ and test MSE. The results again verify the three regimes discussed in Section 6. For both data sizes, models over-compress features when $ID_H < ID_Y$ and thus lead to increasing test MSE (and thus poor generalization). Then, for small datasets with 1K samples, $ID_H \approx ID_Y$ identifies the sweet spot where test MSE tends to be the lowest and exhibits the 'U-shape' plots. Note that models trained with Hopper datasets have not collapsed yet, so they only exhibit the upper part of the 'U-shape'. Finally, with more samples, e.g., 20K, $ID_H \gg ID_Y$ achieves the best generalization with under-compressed models.

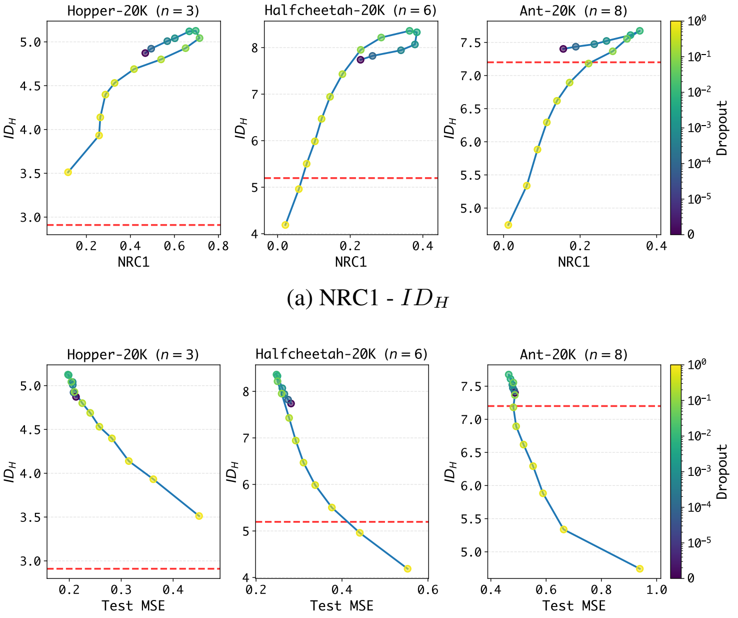
In summary, dropout regularization offers an alternative approach to adjusting the degree of model collapse, while all conclusions drawn from the main body remain intact and inclusive. In addition, mild dropout regularization is more effective than weight decay regularization in increasing NRC1 and $ID_H$ metrics for the under-compressed regime.



(a) NRC1 - $ID_H$

(b) Test MSE - $ID_H$

Figure 13: Relationship between $ID_H$ and NRC1 and Test MSE for Hopper-1K, Halfcheetah-1K, and Ant-1K datasets, when applying model dropout regularization. The horizontal red dashed line represents $ID_Y$.



(a) NRC1 - $ID_H$

(b) Test MSE - $ID_H$

Figure 14: Varying model dropout regularization, relationship between $ID_H$ and NRC1 and Test MSE for Hopper-20K, Halfcheetah-20K, and Ant-20K datasets. The horizontal red dashed line represents $ID_Y$.

### G.2 Model Depth

With mild weight decay regularization, we find that increasing model depth leads to smaller NRC1 and $ID_H$ and thus to more collapsed features. In Figure 15, we examine the relationship between $ID_H$ and NRC1 for Hopper-20K and Halfcheetah-20K datasets. We fix the model width to be 256 and vary the model depth ($\in \{2, 3, 4, 5\}$). For each model depth, we show five mild weight decay values: 0.0001, 0.0003, 0.0005, 0.0007, 0.001. The figure shows that increasing model depth gradually pushes points to the region on the bottom left, where both NRC1 and $ID_H$ are small. For example, Halfcheetah-20K with a depth of 5 can result in collapsed models with $ID_H < ID_Y$.

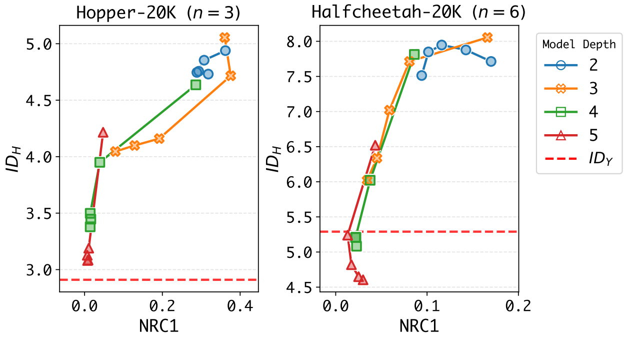


Figure 15: Relationship between $ID_H$ and NRC1 for Hopper-20K and Halfcheetah-20K datasets, when varying model depth. All models have a hidden size of 256. The horizontal red dashed line represents $ID_Y$.

## H Empirical Analysis on More Tasks

We extend our empirical analysis to more challenging tasks with varying data sizes, increased intrinsic dimensions, and visual inputs.

Table 5: Overview of additional datasets employed in this section.

| Dataset | Data Size | Input Type | Input Dim ($D$) | Input ID ($ID_X$) | Target Dim ($n$) | Target ID ($ID_Y$) |
|---|---|---|---|---|---|---|
| Humanoid | 500,000 | raw state | 348 | 11.02 | 17 | 9.85 |
| Relocate | 500,000 | raw state | 39 | 6.90 | 30 | 19.82 |
| Cheetah_run | 80,000 | RGB image | 84 × 84 × 9[1] | 8.26 | 6 | 6.00 |
| Humanoid_walk | 100,000 | RGB image | 84 × 84 × 9 | 9.41 | 21 | 14.84 |

**Humanoid (MuJoCo locomotion)** The Humanoid dataset (Younis et al., 2024) is generated from the MuJoCo physics simulator, introduced in the main body and Appendix A.1. Each state consists of high-dimensional proprioceptive information, and the corresponding targets are the expert control torques applied at each joint. The goal is to enable the humanoid to run forward stably while maintaining balance, which is substantially more difficult than all previously considered MuJoCo tasks due to its high degrees of freedom and complex contact dynamics. Among all MuJoCo environments, Humanoid is widely regarded as the most challenging.

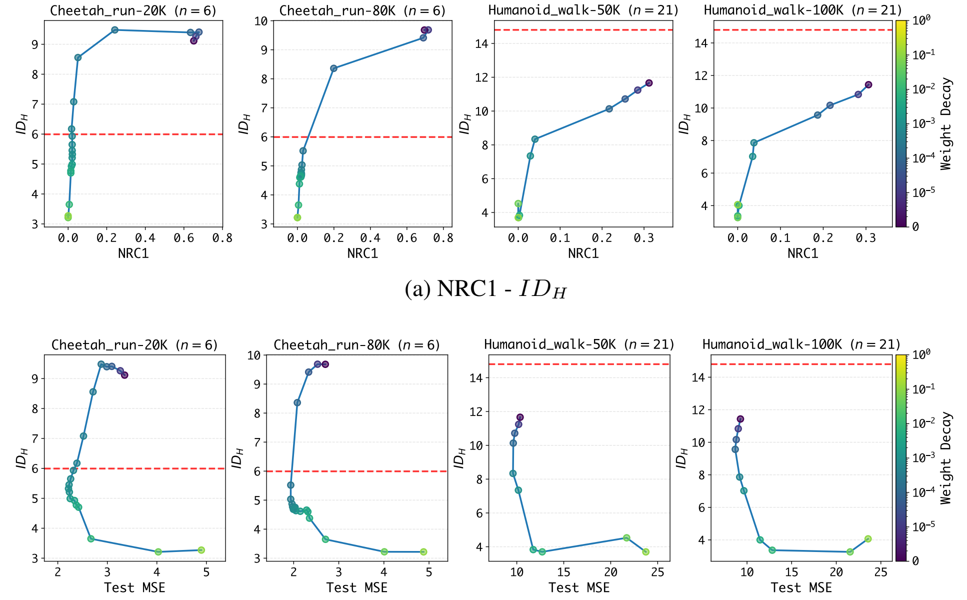
**Relocate (Adroit manipulation)** The Relocate dataset (Fu et al., 2020) comes from the Adroit suite (Rajeswaran et al., 2018) of dexterous manipulation tasks. Adroit uses a simulated 24 degrees of freedom (24-DoF) robotic hand, combined with an arm of up to 6-DoF, with rich contact and articulation dynamics. In the Relocate task, the state includes joint positions, velocities, hand pose information, and kinematic information about the ball and target. The action corresponds to the joint torques for the 24 actuators and to the arm movement. The objective is to grasp a small object and

[1] Frame stack is commonly applied for visual control tasks. A single observation is of shape 84 × 84 × 3. A frame stack of 3 is used in our experiments, resulting in visual inputs of dimension 84 × 84 × 9.

relocate it to a specified target position. It is a long-horizon manipulation task requiring precise coordination and contact control. Among the four Adroit tasks in the D4RL benchmark(Fu et al., 2020), Relocate is widely considered the most difficult due to its combination of dexterity, precision, and exploration complexity.

**Cheetah_run & Humanoid_walk (Visual continuous control)** The Cheetah_run and Humanoid_walk datasets (Lu et al., 2023) are visual control benchmarks constructed from demonstrations generated in the DeepMind Control Suite (Tassa et al., 2018). Inputs consist of raw image observations (e.g., 84×84x3 RGB frames), and targets correspond to the continuous control commands. In this section, the Cheetah_run dataset consists of expert demonstrations, as is the case for all previous datasets, while the Humanoid_run dataset contains some noisy suboptimal behavior in addition to the expert demonstrations ('medium-expert dataset'). This adds difficulty in extracting a good policy by imitating the dataset's behavior. We use a CNN image encoder (consisting of 4 `conv2d` layers and ReLU activation), followed by a 3-layer MLP policy network.

Visual control is particularly interesting and challenging because each frame provides only partial information about the system state, effectively forming a Partially Observed Markov Decision Process (POMDP) (Yarats et al., 2022). As a result, the model must infer underlying transition dynamics and identify salient visual features directly from high-dimensional pixel inputs. Within the visual D4RL tasks (Lu et al., 2023), Humanoid_walk is the most challenging due to the humanoid's instability and high-dimensional dynamics, whereas Cheetah_run is comparatively easier but still nontrivial.



(a) NRC1 - $ID_H$

(b) Test MSE - $ID_H$

Figure 16: Relationship between $ID_H$ and NRC1 and Test MSE for Cheetah_run-20K, Cheetah_run-80K, Humanoid_walk-50K and Humanoid_walk-100K. The horizontal red dashed line represents $ID_Y$.

### H.1 Visual Control Tasks

Figure 16(a) shows the relationship between $ID_H$ and NRC1 for the visual control datasets with varying sizes. Consistent with the conclusions in Section 5, $ID_H$ provides a more refined geometric structure than NRC1. Collapsed models with small NRC1 values have varying $ID_H$ below or in the vicinity of $ID_Y < n$, while non-collapsed models with non-trivial NRC1 maintain their $ID_H$ to be above $ID_Y$ and to be positively correlated with NRC1. Notably, for the most challenging Humanoid_walk task, which has a substantially higher $ID_Y$ due to its complex high-dimensional dynamics, models trained with *zero* weight decay initially exhibit neural regression collapse with $ID_H < ID_Y$. This explains the extremely large test MSE observed for Humanoid_walk in Figure 16(b) and its negative correlation with $ID_H$, which matches the over-compressed regime ($ID_H < ID_Y$) summarized in Table 2. In comparison, the two Cheetah_run datasets exhibit the ‘U-shape’ in the relationship between $ID_H$ and test MSE. This emphasizes the sweet spot with $ID_H \approx ID_Y$, which approaches the lowest test MSE in noisy and challenging tasks.

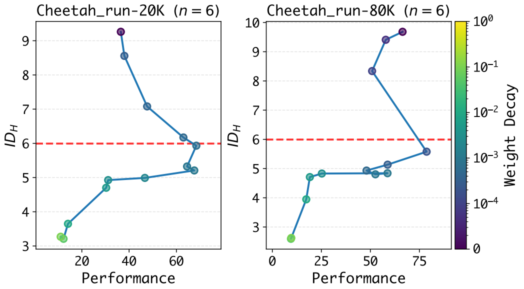


Figure 17: Relationship between $ID_H$ and normalized model performance ($\in [0, 100]$; the higher, the better) trained on Cheetah_run-20K, and Cheetah_run-80K. The horizontal red dashed line represents $ID_Y$.

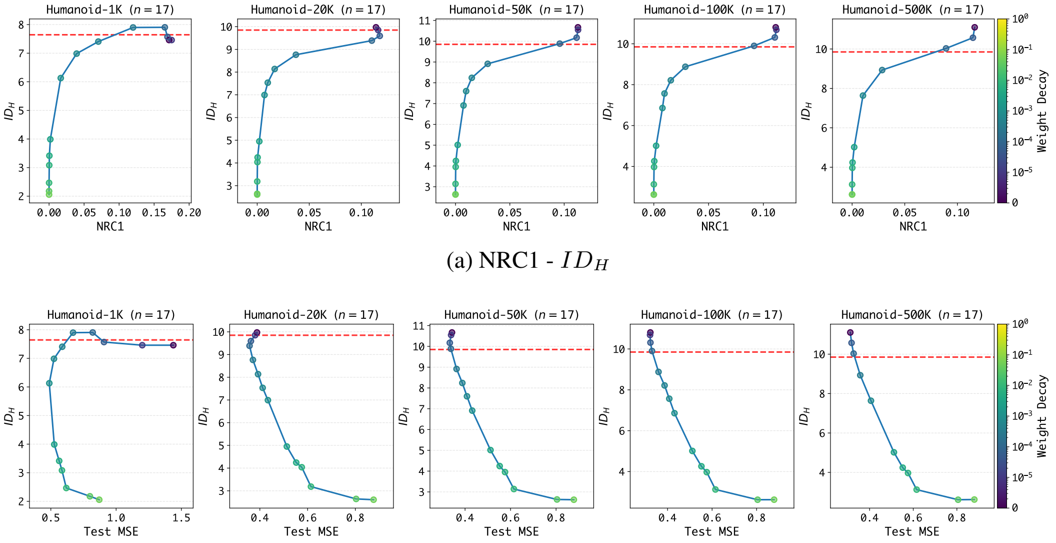
**Practical guideline in control tasks** Our empirical results have direct implications for evaluating real-world control models, where evaluation is often expensive, unsafe, and risky (Levine et al., 2020). Validation MSE provides little indication of actual control performance in the real tasks. Practitioners must therefore rely on costly real-environment interactions to tune hyperparameters and assess the policy. In Figure 17, we show that the relationship between $ID_H$ and $ID_Y$ not only predicts test MSE but also aligns with the *true* control performance obtained from environment interaction. In particular, the relationship between $ID_H$ and normalized model performance exhibits the opposite ‘U-shape’ behavior, achieving its best score when $ID_H \approx ID_Y$, mirroring the sweet-spot identified in Section 6.

In this way, intrinsic dimension provides a lightweight surrogate for policy evaluation, reducing reliance on frequent, costly, and potentially unsafe real-environment testing. During training, one can monitor the relationship between $ID_H$ and $ID_Y$ to narrow down the hyperparameter search by avoiding the $ID_H < ID_Y$ scenario. When $ID_H \geq ID_Y$, the correlation between $ID_H$ and the test MSE can also reflect its correlation with the real performance.

### H.2 RELOCATE & HUMANOID

**Humanoid** In Figure 19, we show ‘NRC1 - $ID_H$’ and ‘Test MSE - $ID_H$’ plots for Humanoid datasets with varying sizes. In Figure 19(a), we observe that $ID_H$ provides a more refined geometric structure than NRC1. Collapsed models with near-zero NRC1 values have varying $ID_H$ below or in the vicinity of $ID_Y < n$, while non-collapsed models with non-trivial NRC1 maintain their $ID_H$ to be above $ID_Y$ and to be positively correlated with NRC1. Then, Figure 19(b) again verifies the three regimes discussed in Section 6. For all data sizes, models over-compress features when $ID_H < ID_Y$ and thus lead to increasing test MSE (and thus poor generalization). Then, for small datasets with 1-50K samples, $ID_H \approx ID_Y$ identifies the sweet spot where test MSE tends to be the lowest and exhibits the ‘U-shape’ plots. Finally, with more samples, e.g., 100-500K, $ID_H > ID_Y$ achieves the best generalization with under-compressed models.

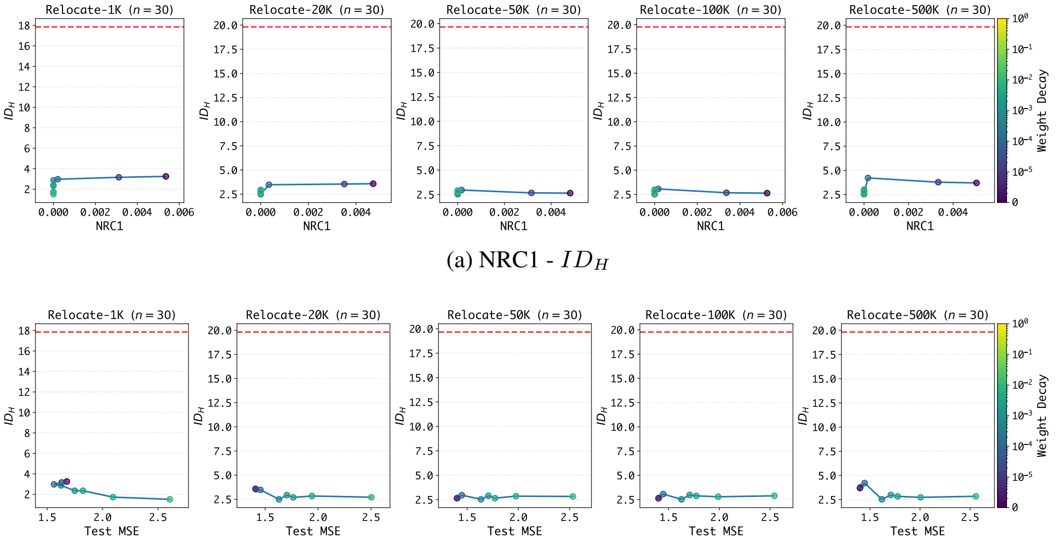
**Relocate** For the Relocate task with varying data sizes, which have the highest $ID_Y$ due to its complex task dynamics, models trained with *zero* weight decay initially exhibit a severe collapse with both NRC1 $\approx 0$ and $ID_H \ll ID_Y$. Unsurprisingly, Test MSE remains large for all data sizes and decay values, and it also negatively correlates with $ID_H$, uncovering the over-compressed regime.



(a) NRC1 - $ID_H$

(b) Test MSE - $ID_H$

Figure 18: Relationship between $ID_H$ and NRC1 and Test MSE, for Humanoid locomotion task, when varying data size ($\in \{1\text{K}, 20\text{K}, 50\text{K}, 100\text{K}, 500\text{K}\}$). The red dashed line represents $ID_Y$.



(a) NRC1 - $ID_H$

(b) Test MSE - $ID_H$

Figure 19: Relationship between $ID_H$ and NRC1 and Test MSE, for Relocate manipulation task, when varying data size ($\in \{1\text{K}, 20\text{K}, 50\text{K}, 100\text{K}, 500\text{K}\}$). The red dashed line represents $ID_Y$.