REPRESENTATIONS HAVE TRAJECTORIES: CHARACTERIZING TEMPORAL REPRESENTATION DYNAMICS IN NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent work has shown that deep neural networks exhibit an implicit low-rank simplicity bias. In this paper, we investigate the questions that naturally arise from this observation. What are the temporal consequences of low-rank simplicity bias during optimization? How does representation geometry evolve throughout optimization rather than only at convergence? Do different neural architectures exhibit distinct geometric optimization behaviors? To answer these questions, we conduct a large-scale empirical study tracking representational geometry throughout training. We find that neural networks follow three recurring geometric trajectories: persistent high-rank, collapse-and-recovery, and persistent low-rank. We further identify Destructive Representation Collapse (DRC) as a pathological regime in which representations collapse to rank-one and fail to recover, leading to a sustained loss of representational capacity. While collapse-and-recovery is consistently associated with successful optimization in controlled settings, experiments on real-world models show that the relationship between geometric trajectories and optimization outcomes is configuration dependent. Overall, our findings suggest that representation geometry evolves dynamically during training and provide a new perspective on how deep neural networks learn.

1 INTRODUCTION

Deep neural networks (DNNs) learn task-relevant representations through nonlinear transformations that evolve throughout optimization (Bengio et al., 2013; Saxe et al., 2014; Baratin et al., 2021). This evolution, termed *representation dynamics*, offers insight into how networks learn beyond their final representations.

Prior work shows that DNNs exhibit an implicit low-rank bias (Huh et al., 2023), and that under suitable conditions, finite-depth networks can recover the target function's rank (Jacot, 2023). However, existing studies largely characterize representations at convergence, leaving their evolution during optimization less understood. We therefore ask: *How does representation geometry evolve during optimization, do architectures exhibit distinct trajectories, and are these trajectories associated with optimization success or failure?*

Contributions. We investigate these questions and make the following observations. In controlled settings, runs can be partitioned into recurring geometric trajectory archetypes. These trajectories include persistent high-rank behavior, transient representation collapse followed by recovery, and persistent low-rank behavior. The prevalence of these trajectories varies considerably across different architectures, optimization settings, and activation functions. To characterize the pathological extreme of persistent low-rank behavior, we introduce Destructive Representation Collapse (DRC): a persistent rank-one state that, in our experiments, is already present at the first logged epoch and is not escaped during training. Our main contributions are summarized as follows.

1. A large-scale empirical study of temporal representation geometry during optimization.

2. Identification and characterization of three recurring rank trajectories.

3. Introduction of DRC, with evidence that it is present early, grows with depth, and is rarer with residual connections.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

4. Statistical evidence that trajectory types are associated with optimization outcomes at convergence.

5. Evidence that trajectory category at early training milestones forecasts eventual optimization success, replicated across synthetic MLPs.

6. CIFAR-10 experiments across MLP, CNN, ResNet, and ViT architectures, replicating the synthetic taxonomy across four architecture families.

2 Related Works

Implicit Low-Rank Bias: Gunasekar et al. (2017) showed that gradient descent in matrix factorization implicitly favors low nuclear-norm solutions, while Arora et al. (2019) showed that depth strengthens this low-rank bias. Huh et al. (2023) found that DNNs favor lower-rank representations and can exploit this bias without increasing capacity. Razin & Cohen (2020) showed that implicit regularization in deep matrix factorization cannot be reduced to simple norm minimization, and Jacot (2023) showed that finite-depth networks can recover the target function's bottleneck rank. These works focus primarily on low-rank solutions or the final rank rather than its evolution during training.

Representation Geometry: Papyan et al. (2020) showed that deep networks develop structured representations through Neural Collapse, while Baratin et al. (2021) linked implicit regularization to dynamic alignment with task-relevant directions. Súkeník et al. (2024) showed that regularized low-rank solutions can achieve lower loss than DNC, while Garrod & Keating (2026) found that DNC persists under cross-entropy as a stable critical point with a larger optimization basin. Saxe et al. (2014) showed that deep linear networks follow nonlinear learning trajectories with long plateaus and rapid convergence. Unlike Dohare et al. (2024), which studies continual learning degradation, we examine temporal representation dynamics and characterize DRC, a persistent rank-one state associated with optimization failure.

Existing work links DNN implicit bias to low-rank representations, neural collapse, and structured feature geometry, but largely studies final representations or optimization behavior rather than temporal evolution of representation. We address this gap through a layer-wise study of $4{,}084$ controlled synthetic runs, identifying trajectory archetypes missed by endpoint analyses. Representation collapse has also been studied in self-supervised learning (Jing et al., 2022), GNN over-smoothing (Li et al., 2018; Oono & Suzuki, 2020), and Transformer attention collapse (Dong et al., 2021). Unlike these settings, we study DRC as a persistent rank bottleneck during supervised training associated with optimization failure.

Rank contraction in supervised networks has likewise been linked to depth: Daneshmand et al. (2020) found stable-rank collapse at initialization in unnormalized networks, prevented by batch normalization, while Feng et al. (2022) observed decreasing feature and Jacobian rank across layers. Signal-propagation analyses (Poole et al., 2016; Schoenholz et al., 2017; Pennington et al., 2017) describe ordered and chaotic regimes through the evolution of input correlations; our DRC regime is closely related to the ordered phase, appearing from the first logged epoch and becoming more prevalent with depth. Complementarily, we track full training trajectories, distinguishing runs that escape low-rank states from those that exhibit persistent collapse, and relate these trajectories to optimization outcomes across architectures and optimizers. Although effective rank has been used to assess representation quality and downstream performance (Garrido et al., 2023), we use it as a training-time criterion for identifying collapse.

3 Preliminaries and Experimental Design

3.1 Neural Networks and Over-parameterization

Consider an L-layer neural network ($l \in [1, L]$) with input $\mathbf{x} \in \mathbf{R}^{d_0}$. The hidden representations are recursively defined $\mathbf{h}^{(0)} = \mathbf{x}, \quad \mathbf{h}^{(l)} = \sigma\big(\mathbf{W}^{(l)}\mathbf{h}^{(l-1)}\big)$

The network output is given by, $\mathbf{y} = \mathbf{W}^{(L)}\mathbf{h}^{(L-1)}$, where $\mathbf{W}^{(l)}$ denotes the weight matrix of the l-th layer and $\sigma(\cdot)$ is a nonlinear activation function (e.g., ReLU). Setting $\sigma(\cdot)$ to the identity map

yields a linear network. Throughout this work, we consider overparameterized networks in which the number of trainable parameters exceeds the number of training samples. ($N_{samples} = 32 \ll N_{parameters}$) with d_{in} and d_{out} = 64, base hidden dimension=256.

3.2 Teacher-Student Framework

As for synthetic benchmark, we consider a teacher-student regression setting (Saxe et al., 2014) in which the student network learns a target function generated by a teacher matrix $W_{teacher} \in \mathbf{R}^{d_{out} \times d_{in}}$ with controllable rank r, where $r \in \{1, \ldots, \min(d_{\text{out}}, d_{\text{in}})\}$. The teacher matrix is constructed as $W_{teacher} = U\Sigma V^{\top}$, where $U \in \mathbf{R}^{d_{out} \times r}$ and $V \in \mathbf{R}^{d_{in} \times r}$ are orthonormal matrices sampled via QR decomposition, r denotes the target rank, and $\Sigma \in \mathbf{R}^{r \times r}$ contains diagonal singular values. Training inputs are sampled independently from standard Gaussian distribution, $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and the corresponding targets are generated as $\mathbf{y} = W_{teacher}\mathbf{x} + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \tau^2 I)$ with $\tau = 0.0$. We employ controlled experiments to isolate the effects of dataset types, task ranks, architecture, initialization, data augmentation, and optimization. Controlled, isolated experiments help us study the representation geometry clearly, without noise artifacts.

3.3 Representation Geometry

Stable Rank. (Rudelson & Vershynin, 2007) measures how dominant the single strongest direction is relative to the total spectral energy. It is the ratio of the squared Frobenius norm to the squared spectral norm of the activation matrix: $\operatorname{srank}(\mathbf{H}^{(\ell)}) = \frac{\|\mathbf{H}^{(\ell)}\|_F^2}{\|\mathbf{H}^{(\ell)}\|_2^2}$

Effective Rank. (Roy & Vetterli, 2007) measures how many directions carry meaningful mass across the entire spectrum via entropy, capturing representational diversity that becomes hard for stable rank to capture once the tail is small relative to σ_1. Computed as the exponential of the Shannon entropy of the normalized singular values of the activation matrix: $\operatorname{erank}(\mathbf{H}^{(\ell)}) = \exp(-\sum_i p_i \ln p_i)$, where $p_i = \frac{\sigma_i}{\sum_j \sigma_j}$. Layer activations $H^{(\ell)}$ are computed on the train_eval loader, an unaugmented, deterministic evaluation subset of the training split; preprocessing follows the released code

We report both the stable and effective ranks because they are sensitive to structurally distinct features of the singular value spectrum. Stable rank is anchored to spectral norm, so it is particularly sensitive to the dominance of a single leading direction, but is comparatively insensitive to how mass is distributed across the remaining spectrum once that mass is small relative to σ_1. Effective rank is sensitive to how many directions carry non-trivial mass in aggregate, but is less sensitive to extreme dominance by a single direction as long as some distributed tail persists.

3.4 Geometric Trajectory Analysis

Stable and Effective ranks are computed across all layers of a network and at specific epochs, such as $\{1, 10, 50, 100, 200, \ldots, T\}$. Ranks are tracked at two different critical locations of the network. (1) **Min-layer:** The minimum rank observed across all layers at a given epoch. This represents the primary dimensional bottleneck, since it is the narrowest path of the representational pipeline. It is our primary metric for synthetic runs. (2) **Penultimate-layer:** The rank observed at the second-to-last hidden layer. It is reported to be compatible with prior endpoint-based studies of representational collapse (Huh et al., 2023).

Dataset: As for **Data Generation**, our input ($\mathbf{X}$) is a high-dimensional white noise: $\mathbf{X} \sim \mathcal{N}(0, I) \in \mathbf{R}^{\mathrm{N} \times 64}$ (one sample per row) and the target is $\mathbf{Y} = \mathbf{X}W_{\text{teacher}}^{T}$ where $\mathbf{X} \in \mathbf{R}^{\mathrm{N} \times 64}$.

Our **experimental setup in sweeps** is as follows. The input/output dimension (d_{in}, d_{out}) is constant at 64. The Task Rank (r) is swept across $\{1, 4, 16, 24, 32, 64\}$. The training sample size (N_{train}) is held constant at 32 to keep the model in an over-parameterized regime. Our **data splits and sizes** are as follows. The train split is 32 samples. The validation split is $max(64, \mathrm{N}_{train})$. The test split is $max(256, \mathrm{N}_{train})$. We use a seed value of 42 for regression and cross-validation analysis. Information regarding *Architectures*, *Optimizers*, *Initialization schemes*, *Activation functions*, *Residual Connections* used across sweeps can be found in the Appendix Sec. I.

4 Discovery of Geometric Trajectory Archetypes

Tracking representation geometry throughout optimization shows that runs can be partitioned into recurring geometric trajectory archetypes according to the operational criteria in Table 1. These trajectories differ in how the representation geometry evolves over the course of optimization. Figure 1 visualizes examples.

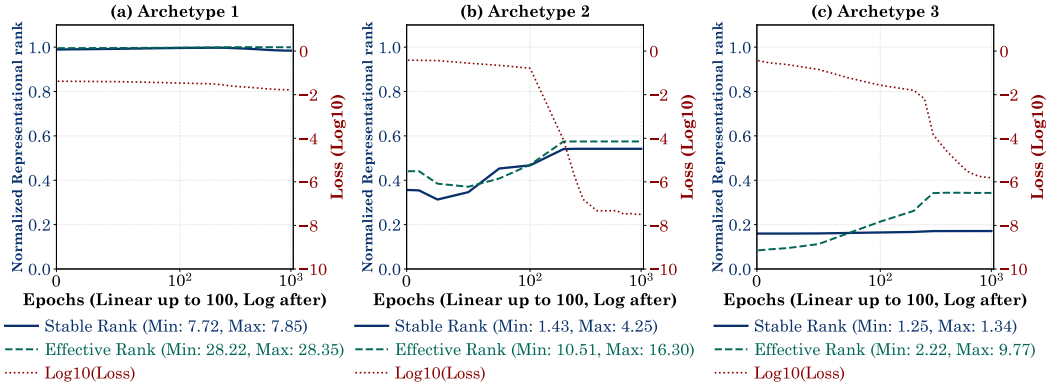


Figure 1: **Archetypal representation geometry trajectories:** (a) persistent high rank, (b) collapse-and-recovery, and (c) persistent low rank. The x-axis uses a symlog scale to resolve early dynamics.

Archetype 1 (A1): Persistent High Rank. Throughout optimization, both stable and effective rank remain consistently high with minor fluctuations, indicating that representations preserve their high-dimensional geometry throughout training.

Archetype 2 (A2): Collapse and Recovery. A2 runs are those in which the stable rank reaches or drops below the collapse threshold of 1.5 at some logged epoch and later rises above it by more than 0.1. In the example shown in Figure 1(b), this recovery is accompanied by a clear drop in loss. A2 behavior occurs under both default and low-rank initialization (Table S.5).

Archetype 3 (A3): Persistent Low Rank. Representations enter a low-rank regime early in optimization and remain there for the rest of training. Unlike Archetype 2, little or no recovery in representational rank is observed afterward.

DRC: Destructive Representation Collapse. Within the persistent low-rank regime, we identify an extreme case in which representations are effectively one-dimensional and remain so. Because stable rank alone cannot distinguish moderately low-rank representations from complete collapse, DRC is defined jointly using stable and effective rank (Eq. 1). In our experiments, most DRC runs are already at SR ≤ 1.05 at the first logged epoch (Section 7), so DRC is better understood as a failure to escape an early rank-one state than as a collapse that develops later in training.

Table 1: **Rank dynamics archetypes**. Here, s_t denotes stable rank at epoch t, $s_{\min} = \min$ over logged epochs $t \in 1, 10, 50, 100, \ldots, T$ of s_t. The same notation holds for effective rank e_t as well.

Archetype	Criterion
A1 (Persistent High-Rank)	$\text{SR}_t > 1.5$ at every logged epoch t
A2 (Transient Collapse and Recovery)	$(\text{SR}_{\min} \leq 1.5) \wedge (\text{SR}_{\text{final}} > 1.5) \wedge (\text{SR}_{\text{final}} - \text{SR}_{\min} > 0.1)$
A3 (Persistent Low-Rank)	$(\text{SR}_{\min} \leq 1.5) \wedge [\text{SR}_{\text{final}} \leq 1.5 \vee (\text{SR}_{\text{final}} - \text{SR}_{\min} \leq 0.1)] \wedge (\text{ER}_{\text{final}} > 1.5)$
DRC (Destructive Representation Collapse)	$(\text{SR}_{\min} \leq 1.05) \wedge (\text{SR}_{\text{final}} \leq 1.05) \wedge (\text{ER}_{\text{final}} \leq 1.5)$

Together, these archetypes provide a temporal view of representation dynamics during optimization. Across the $4{,}084$ runs, A1 was the most common trajectory ($1{,}746$, 42.75%), followed by DRC ($1{,}106$, 27.08%), A3 ($1{,}045$, 25.59%), and A2 (187, 4.58%). A2 had the lowest mean and median losses (0.0580 and 0.00001), while DRC had a median loss of 0.3676 and a mean loss of 1.05×10^{15} due to divergence. The corresponding mean stable ranks were 7.8135, 2.7860, 1.1748, and 1.00 for A1–A3 and DRC, respectively. This taxonomy captures differences not only in how representation geometry evolves, but also in the optimization outcomes associated with these trajectories.

The observed trajectory patterns are also stable across different initialization seeds. We repeated the full $2{,}268$-configuration factorial grid using two additional seeds $(43$ and $44)$, resulting in $6{,}804$ fully trained runs across all three seeds. The DRC fraction remained stable across seeds

$(41.80\%, 41.23\%$, and $42.46\%)$. It differs from the 27.08% reported above, which is over the full 4,084-run pool that also contains the residual sweeps. The association between recovery trajectories and optimization success was also consistent across all three seeds, indicating that the reported geometric dynamics are not specific to the original initialization. (See Table S.1 for complete details).

5 Statistical Significance of Geometric Trajectories

We next examine whether the discovered trajectory archetypes are associated with optimization outcomes. Across all convergence thresholds, Collapse-and-Recovery (A2) trajectories consistently show the highest success rates, whereas DRC exhibits the lowest. At the strict 10^{-3} threshold, optimization success reaches 82.89% for A2 but only 0.90% for DRC; A1 and A3 lie between these extremes at 35.11% and 40.96%, respectively.

At this threshold, recovery trajectories are $3.07\times$ more likely to optimize successfully than non-recovery trajectories. We tested this association using Fisher's exact test (Fisher, 1922) and examined whether it persists after controlling for architectural and optimization factors. Archetypes are defined from the trajectory, including its final epoch, so these are associations rather than prospective effects (Table 2).

Table 2: Association between the recovery trajectory (A2) and optimization success at three loss thresholds ($M_{\text{models}} = 4{,}084$, seed 42). Contingency counts, RR (Relative Risk) and the unadjusted OR (Odd's Ratio) compare A2 with all other runs (A1+A3+DRC); the unadjusted OR uses the Haldane–Anscombe correction ((a+0.5)(d+0.5))/((b+0.5)(c+0.5)) with Fisher's exact p. The adjusted OR is the A2 coefficient of the multivariate logistic regression (adjusting for activation, architecture, depth, task rank and optimizer; not initialization) and therefore compares A2 with A1 only. a: A2+success; b: A2+failure; c: non-A2+success; d: non-A2+failure.

Metric	**Loss** < 0.05	**Loss** $< 10^{-2}$	**Loss** $< 10^{-3}$
Contingency (a, b, c, d)	$(164, 23, 1517, 2380)$	$(156, 31, 1132, 2765)$	$(155, 32, 1051, 2846)$
Success Rate (A2 vs. No A2)	87.70% vs. 38.93%	83.42% vs. 29.05%	82.89% vs. 26.97%
RR [95% CI]	2.25 [$2.11, 2.41$]	2.87 [$2.65, 3.11$]	3.07 [$2.83, 3.34$]
Unadjusted OR [95% CI]	10.98 [$7.09, 17.00$]	12.13 [$8.22, 17.90$]	12.95 [$8.82, 19.03$]
Fisher p**-value**	1.07×10^{-41}	4.74×10^{-51}	2.67×10^{-54}
Adjusted OR [95% CI]	**6.77** [$3.92, 11.70$] $p < 0.0001$	**4.41** [$2.40, 8.08$] $p < 0.0001$	**4.59** [$2.39, 8.81$] $p < 0.0001$

The association is statistically significant (Fisher's exact test, $p = 2.67 \times 10^{-54}$, unadjusted OR = 12.95). In a multivariate logistic regression adjusting for architecture, optimizer, activation function, depth and task rank (Table 3), A2 runs have $4.59\times$ higher odds of success than A1 runs ($p < 0.0001$), this comparison is against A1, not against the pooled non-recovery group. Trajectory type is therefore associated with success beyond these configuration factors. Because archetypes are defined based on the trajectory, including their final epoch, we do not interpret this as a causal or prospective effect. A separate analysis restricted to information available before convergence (Appendix Sec. E) supports a genuinely prospective claim, see Sec. 6.1. A3 has higher unadjusted success than A1 $(40.96\%$ vs. $35.11\%)$ but lower adjusted odds (OR 0.29).

6 Factors Influencing Geometric Trajectories

To understand how architectural and hyperparameter choices influence geometric trajectories, we evaluated $M_{\text{models}} = 4{,}084$ synthetic models across different depths, task ranks, architecture classes, activations and optimizer types.

6.1 Multivariate analysis.

Univariate comparisons can be influenced by confounding factors, as architectural choices (e.g., residual connections or Adam optimization) affect both convergence and representational rank. To isolate the independent effect of each factor on optimization success ($\mathcal{L}_{\text{train}} < 10^{-3}$), we fitted a multivariate logistic regression model including all hyperparameters simultaneously: $\text{logit}(P(Y = 1)) = \beta_0 + \sum_j \beta_j X_j$. The initialization scheme is not a covariate in this model. From Table 3 and Fig. 2, we note that:

1. Representation Archetypes: After controlling for architecture, optimizer, activation, depth, and task rank, models following the recovery trajectory (A2) have $4.59\times$ higher odds of successful

Table 3: **Multivariate logistic regression of factors influencing optimization success ($\mathcal{L} < 10^{-3}$, $M_{\text{models}} = 4{,}084$).** Reference baselines: Archetype = A1, Activation = GELU, Architecture = Plain MLP, Optimizer = Adam, Seed = 42, Initialization Scheme is not a covariate. Archetype is defined by the trajectory, so the coefficients of the other factors are conditional on the archetype.

Factor / Variable	Coef (β)	SE	Adjusted OR [95% CI]	p-value
Intercept	$+2.0358$	0.4662	$7.66\ [3.07, 19.10]$	< 0.0001
Archetype A2	$+1.5229$	0.3334	**4.59** $[2.39, 8.81]$	< 0.0001
Archetype A3	-1.2377	0.1862	$0.29\ [0.20, 0.42]$	< 0.0001
DRC	-5.2012	0.3896	**0.0055** $[0.0026, 0.0118]$	$< 10^{-20}$
Residual MLP	$+1.6988$	0.3917	$5.47\ [2.54, 11.78]$	< 0.0001
Deep Linear MLP	$+0.4347$	0.3881	$1.54\ [0.72, 3.30]$	0.2627
Model Depth (L)	-0.0136	0.0036	$0.987\ [0.979, 0.994]$	0.0002
Task Rank (r)	-0.0166	0.0036	$0.984\ [0.977, 0.990]$	< 0.0001
Momentum SGD	-5.1735	0.1996	$0.0057\ [0.0038, 0.0084]$	$< 10^{-20}$
Standard SGD	-6.5178	0.3954	$0.0015\ [0.0007, 0.0032]$	$< 10^{-25}$
Leaky ReLU	$+0.3987$	0.2802	$1.49\ [0.86, 2.58]$	0.1547
SELU	$+0.2707$	0.2982	$1.31\ [0.73, 2.35]$	0.3640
Tanh	$+0.2423$	0.3025	$1.27\ [0.70, 2.31]$	0.4231
ReLU	$+0.1908$	0.2716	$1.21\ [0.71, 2.06]$	0.4824
Linear	-0.2478	0.3067	$0.78\ [0.43, 1.42]$	0.4191
SIREN	-2.8662	0.2885	$0.057\ [0.03, 0.10]$	< 0.0001

Coef (β) represents the fitted change in log-odds. **SE** measures the coefficient standard error via Hessian inversion. **Adjusted OR** (e^{β}) quantifies the factor's independent odds multiplier after controlling for all other covariates. p**-value** indicates Wald z-test statistical significance.

convergence than the A1 baseline, whereas DRC nearly eliminates the probability of successful optimization (Table 3).

2. Architecture Class: Residual skip connections substantially improve optimization success, increasing the odds of convergence by $5.47\times$ relative to Plain MLP baselines. Residual MLPs exhibit a much higher proportion of high-rank trajectories (A1 $= 54.50\%$) compared with Plain MLPs (A1 $= 32.33\%$) and Deep Linear models (A1 $= 21.43\%$), while reducing the occurrence of catastrophic DRC from 43.62% (Plain MLPs) to 8.51% (Residual MLPs).

However, once a model enters a low-rank state (SR ≤ 1.5), recovery remains uncommon across architectures, with similar A2 recovery rates (5.28% (residual MLPs) vs. 3.80% (plain MLPs)). Instead of eliminating rank decay, residual connections primarily delay its onset to deeper layers, shifting the mean collapse onset at depth $L = 64$ from layer 3.81 in Plain MLPs to layer 16.21 in Residual MLPs (Fig. S.2). The residual coefficient remains large after adjusting for archetype (OR = 5.47), so residual connections likely also help through mechanisms not captured by our rank metrics.

3. Optimizer Dynamics: Within our experimental setting, Adam is associated with substantially higher optimization success than SGD variants. Standard SGD (Adjusted OR $= 0.0015$, $p < 10^{-25}$) and Momentum SGD (Adjusted OR $= 0.0057$, $p < 10^{-20}$) reduce optimization success odds by 99.85% and 99.43% relative to Adam.

4. Depth (L) and Task Rank (r): Each additional layer of depth reduces success odds by 1.35% (Adjusted OR $= 0.9865$, $p = 0.0002$). Each additional unit of teacher rank reduces success odds by about 1.6% (Adjusted OR $= 0.9836$, $p < 0.0001$)

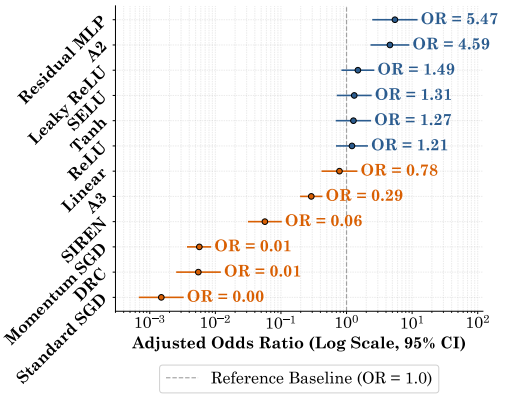


Figure 2: Multivariate Forest Plot: Controlled Predictors of Optimization Success.

5. Activation Geometry: Piecewise-linear and smooth monotonic activations (Leaky ReLU, SELU, Tanh, and ReLU) perform similarly to the GELU baseline, with no statistically significant

difference in optimization success (Adjusted OR $= 1.21$–1.49, $p > 0.15$). In contrast, periodic SIREN activations are associated with substantially lower convergence success overall, reducing the odds of successful convergence by $\sim 95\%$ after controlling for optimizer effects. The SGD-specific failure of SIREN activations is examined separately in Sec. 8.2 (Fig. 5). SIREN runs almost never collapse to rank one, yet rarely succeed, so a high rank is not sufficient for optimization success.

Early trajectory forecasting and association with outcomes. Trajectory category provides a strong early signal of eventual optimization success in synthetic MLPs. Using only information available at each milestone, without final-epoch leakage, trajectory categories at $10\%, 25\%$, and 50% of training predict eventual success with odds ratios of $6.33\times$, $10.60\times$, and $12.11\times$, respectively. The classification uses only information available up to the respective milestone and does not use final loss or final representations. These effects remain stable across 20 threshold combinations (Appendix Sec. E, F). The 25% milestone therefore provides a particularly clear early indication of eventual optimization success. For completeness, retrospective association analyses using the full trajectory show that adding trajectory archetypes to hyperparameter controls increases AUC from 0.9459 to 0.9660 (accuracy $90.77\% \rightarrow 92.58\%$), while final-rank metrics alone achieve an AUC of 0.7743 (69.20%) (Table S.9).

7 Destructive Representation Collapse (DRC)

Prior work has shown that deep networks develop low-rank representations and that rank contracts with depth in unnormalized feedforward networks (Sec. 2). In our synthetic experiments, the most severe form of this contraction, an effectively rank-one representation, becomes more frequent with depth in networks without residual connections or depth-dependent scaling (Fig. S.4), varies with activation function (Fig. S.5), and is associated with optimization failure (Figs 3- 4, Tables S.3- S.4). We call this state DRC.

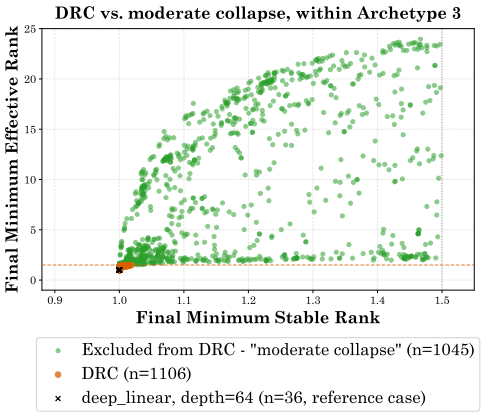


Figure 3: Empirical separation of DRC from moderate low-rank collapse (A3).

Stable rank alone does not identify complete collapse; it is dominated by the leading singular value σ_1 and is insensitive to the rest of the spectrum. A model can have low stable rank (SR ≈ 1.05–1.19) while retaining substantial effective rank (ER ≥ 10.4), as in a Plain CNN run that reaches 90.35% CIFAR-10 test accuracy (Table S.7, Fig. S.6). Hence, we define DRC using effective rank, together with stable-rank conditions (Eq. 1) A DNN comes under the DRC regime if it satisfies the following criterion:

$$\mathbf{SR_{min} \leq 1.05 \ \wedge \ SR_{final} \leq 1.05 \ \wedge \ ER_{final} \leq 1.5} \tag{1}$$

Each component captures a different aspect of collapse. The minimum stable rank indicates whether the representation ever collapses to a nearly one-dimensional state; the final stable rank confirms that it never recovers, and the effective-rank criterion ensures that the remaining singular-value spectrum has also collapsed rather than retaining meaningful high-dimensional structure. Appendix Sec. C justifies the selected thresholds to define DRC. Across the synthetic pool, DRC occurs in $1,106$ of $4,084$ runs (27.08%), but its prevalence varies substantially by architecture: 60.32% of Deep Linear runs, 43.62% of Plain MLP runs, and 8.51% of Residual MLP runs exhibit DRC. Architecture-wise breakdown is reported in Table S.4.

Two observations support the association of DRC with optimization failure.

1. For image classification on CIFAR-10, all 100.0% of models $(41/41)$ that entered the DRC regime collapse to a random guess (Accuracy $\leq 10.15\%$, mean 9.99%), with cross-entropy loss freezing at baseline ≈ 2.3026.

2. In synthetic matrix regression, DRC accounts for 1,106 of 4,084 synthetic runs (27.08%). Crucially, 100.0% **of all numerical loss explosions** $(152/152$ runs with $\mathcal{L} > 100)$ occur exclusively

within the DRC regime, with losses expanding up to 10^{15}. Exploding activations are themselves dominated by a single direction, so these runs are not independent evidence that collapse precedes failure. Among the 954 non-divergent DRC runs, the vast majority fail to optimize, freezing at high unlearned baseline loss (median MSE ≈ 0.29), while only 0.90% of all DRC runs reach $L < 10^{-3}$ (Table S.3)

When DRC arises. In the three-seed grid $(6,804$ runs), all $2,846$ models already have SR ≤ 1.05 at epoch 1. With full-batch training, epoch 1 is one gradient step from initialization, indicating that DRC is present essentially from the start rather than emerging during training, consistent with signal-propagation failure in deep unnormalized networks (Sec. 2). DRC is less prevalent with default initialization (29.3%) than with the rank-one initialization (54.4%) used in half the grid (Table S.5), but remains frequent under default initialization. We therefore interpret DRC as identifying runs trapped in an early rank-one state, not as evidence that training dynamics cause rank collapse. The probability that a model undergoes DRC scales nearly monotonically with architectural depth L. (Fig. S.4).

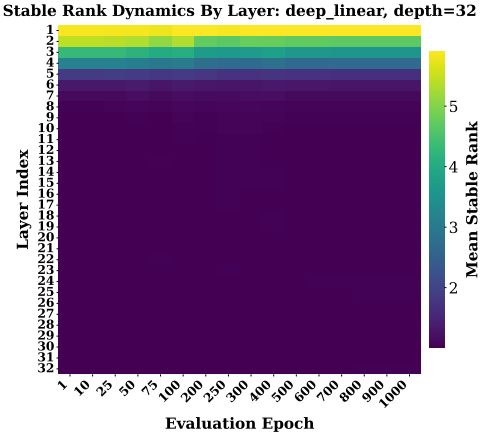


Figure 4: Layer-wise stable rank dynamics: deeper layers undergo DRC without recovery.

Importantly, our results show that low-rank representations should not be viewed as uniformly pathological. Instead, DRC represents an extreme point on the low-rank spectrum that is consistently associated with optimization failure (Fig. 3).

8 CIFAR-10 Experiments

The preceding sections establish the proposed trajectory taxonomy using controlled matrix regression experiments. To test consistency, we now ask which of these patterns appear in 306 CIFAR-10 (Krizhevsky, 2009) vision models, spanning MLP, CNN, ViTs and residual architectures.

8.1 Architecture-Dependent Recovery Trajectories

Across broader CIFAR-10 architectural sweeps, representational trajectories exhibit the following patterns:

Plain MLPs. As in the controlled experiments, models following the recovery trajectory consistently optimize better than those remaining in the persistent low-rank regime. Recovery is associated with lower training loss (0.8380 A2 vs. 2.0072 A3; 1.1692 nats reduction) and test loss (1.5204 A2 vs. 2.1899 A3) (Table S.26). This is consistent with the synthetic results.

Residual MLPs and ResNets. In the CIFAR-10 sweeps, residual architectures reached low training and test loss and did not enter the DRC regime. The 6 ResNet runs (ResNet$-18/50 \times 3$ skip modes) retain high effective rank $(11.58$–$14.45)$ at the end of training. These observations are consistent with skip connections delaying rank decay (Section 6).

Vision Transformers. Across 48 CIFAR-10 ViT runs with and without residual connections, no-residual ViTs showed strong depth-wise rank contraction, with 2 of the 8 depth-9 no-residual runs collapsing under the dual-metric DRC criterion to rank one at chance accuracy, whereas zero residual ViTs collapsed (Table S.10). Early Recovery runs showed substantially higher eventual success than Early Stuck runs, with odds ratios of $23.00\times$, $12.78\times$, and $12.78\times$ at $10\%, 25\%$, and 50% of training, respectively (Table S.12), indicating the geometric framework extends to Vision Transformers; full details are in Appendix Sec. G.

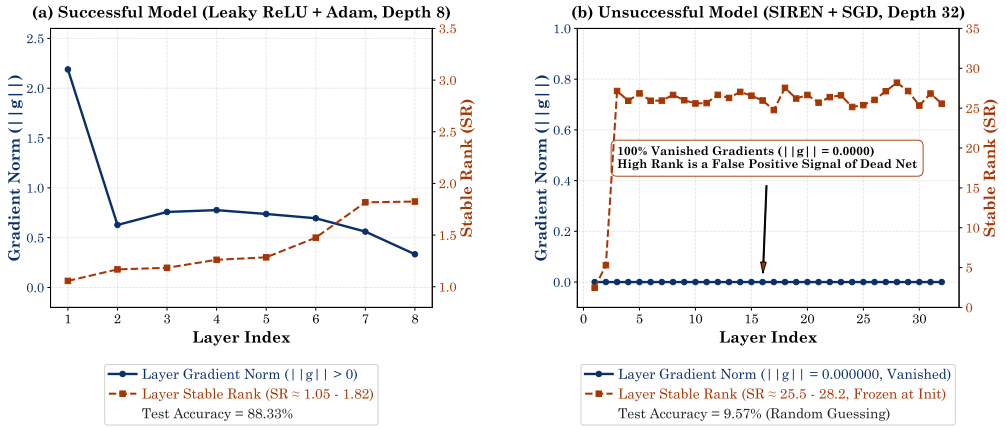


Figure 5: Active learning in a successful model (A) versus vanished layerwise gradients ($\|g\| = 0.0000$) in an un-converged model (B).

8.2 Dead Gradient Effect in Plain Deep CNNs.

In plain feedforward CNNs ($M_{\text{models}} = 192$), a superficial examination suggests an apparent contradiction: models achieving low cross-entropy loss predominantly reside in the low-rank regime ($\text{SR}_{\text{final}} \leq 1.5$), whereas high-rank networks exhibit poor loss optimization. Tracking layerwise gradient norms ($\|g_l\|$) resolves this issue:

1. Dead-Gradient High-Rank scenario: 9.38% of plain CNN runs ($18/192$) remain in Archetype 1. Crucially, 100.0% of these 18 high-rank models failed to optimize, freezing at high test cross-entropy loss (mean $= 2.6445$, range: $2.3353 - 2.7659$) and high training loss (mean $= 2.6550$). As illustrated in Figure 5(b), these runs consist exclusively of deep ($L = 32$) SIREN-activation CNNs under SGD whose layerwise gradient norms vanished to absolute zero ($\|g_l\| = 0.000000, \forall l \in [1, 32]$). In this regime, a high stable rank does not indicate representational capacity in use; it reflects a network that has not been trained.

2. Low-Rank Regularized models: Excluding unconverged dead gradient runs, other plain CNN models operate in Archetype 2-3 and span a wide range of test accuracy (Figure 5(a)).

3. Destructive Collapse (DRC): In contrast, 21.35% of plain CNN runs ($41/192$) satisfy the DRC criterion, all 41 are at chance accuracy with test cross-entropy near the random guess baseline ($-\ln 0.1 \approx 2.3026$, mean 2.4445).

These findings align with Section 7: low-rank representations are not inherently pathological, but the rank-one DRC regime and high-rank runs with vanished gradients are both associated with failure. Rank metrics, therefore, complement rather than replace loss curves and gradient diagnostics. Since DRC is visible at the first logged epoch, monitoring SR and ER could help flag doomed runs early, although we have not tested this as an early-stopping rule.

9 Conclusion & Future Work

Overall, representation trajectories reveal optimization behavior beyond static endpoints: successful runs tend to avoid persistent Destructive Representation Collapse (DRC), while recovery from low-rank states is associated with convergence. Residual connections are associated with fewer DRC cases and delayed rank decay, and trajectory categories at 25% of training already forecast eventual outcomes, suggesting potential prospective value.

Future work includes formal bounds, comparisons with Hessian, edge-of-stability, and plasticity-loss diagnostics (e.g., dormant neurons), activation-centering and stabilization in ViTs, larger per-architecture CIFAR-10 sweeps, higher-order spectral measures such as Schatten norms and singular-value decay, and evaluating early-milestone trajectory categories as practical early-stopping signals.

AI AND LLM DISCLOSURE STATEMENT

In accordance with the ICLR Policy on LLM and AI Use, we disclose that large language models (specifically OpenAI ChatGPT, Google Gemini and Anthropic Claude along with Research engines such as Perplexity and Gemini Deep search) were utilized during the preparation of this manuscript under strict authorial oversight for the following specified tasks:

- **Writing Assistance and Editorial Polishing:** LLMs were employed to improve grammatical precision, sentence clarity, academic tone, and typographical flow, as well as to assist with formatting complex LaTeX tables and multi-panel figures.
- **Literature Retrieval and Discovery:** LLMs were queried to assist in identifying and indexing relevant prior literature across dimensional collapse, matrix factorization dynamics, neural network initialization theory, and transformer representation scaling.
- **Coding and Experimental Assistance:** LLMs were used to assist with writing, debugging, and refining experimental code, including data processing, model implementation, training pipelines, and analysis scripts. All generated or suggested code was reviewed, modified, and validated by the authors before use.

Scope of Human Verification and Exclusions: Large language models were *not* used to generate experimental data, simulate run logs, formulate mathematical proofs, or synthesize scientific conclusions. All empirical runs across multi-seed grids, CIFAR-10 experiments, Singular Value Decompositions (SVD), effective rank/stable rank evaluations, and statistical hypothesis tests were executed on local computational hardware and computed directly from empirical logs. All cited publications, historical attributions, mathematical definitions, and empirical interpretations were verified and cross-checked manually by the authors. The authors retain full accountability and responsibility for the entire scientific content, claims, and integrity of this manuscript.

REPRODUCIBILITY STATEMENT

We describe our experimental setup, including the teacher-student regression framework, data generation process, and definitions of stable and effective rank, in Section 3. The operational criteria used to define the four trajectory archetypes (A1, A2, A3, DRC) are given in Table 1, with the empirical and theoretical justification for the DRC thresholds detailed in Appendix Sec. C. Full architectural specifications (Plain MLP, Deep Linear, Residual MLP, Plain CNN, ResNet-18/50, ViT) and the configuration of every experimental sweep (optimizers, initialization schemes, activation functions, residual connection variants, depths, task ranks, and number of runs) are provided in Appendix Sec. I, Tables S.13– S.29, and a consolidated set of shared hyperparameters (random seed, learning rate, weight decay, batch size, loss criterion) is given in Table S.8. Dataset construction and preprocessing for both the synthetic low-rank regression task and CIFAR-10 are described in Appendix Sec. H. Statistical procedures used to assess significance (odds ratios, adjusted odds ratios via multivariate logistic regression, Fisher's exact test, confidence intervals, and ROC/AUC) are defined in Appendix Sec. D, with the corresponding regression specification in Section 6.1 and Table 3, and the cross-validation protocol in Appendix Sec. E.1 (Table S.9). Robustness of the reported trajectory taxonomy is assessed by repeating the full synthetic grid across three random seeds (Table S.1) and by sweeping the archetype classification thresholds over 20 combinations (Appendix Sec. F). We will release anonymized code for all synthetic and CIFAR-10 experiments, including data generation and training as supplementary material.

REFERENCES

Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/c0c783b5fc0d7d808f1d14a6e9c8280d-Paper.pdf.

Aristide Baratin, Thomas George, César Laurent, R Devon Hjelm, Guillaume Lajoie, Pascal Vincent, and Simon Lacoste-Julien. Implicit regularization via neural feature alignment. In Arindam

Banerjee and Kenji Fukumizu (eds.), *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 2269–2277. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/baratin21a.html.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013. doi: 10.1109/TPAMI.2013.50.

Hadi Daneshmand, Jonas Kohler, Francis Bach, Thomas Hofmann, and Aurelien Lucchi. Batch normalization provably avoids ranks collapse for randomly initialised deep networks. *Advances in Neural Information Processing Systems*, 33:18387–18398, 2020.

Shibhansh Dohare, J Fernando Hernandez-Garcia, Qingfeng Lan, Parash Rahman, A Rupam Mahmood, and Richard S Sutton. Loss of plasticity in deep continual learning. *Nature*, 632(8026): 768–774, 2024.

Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International conference on machine learning*, pp. 2793–2803. PMLR, 2021.

Ruili Feng, Kecheng Zheng, Yukun Huang, Deli Zhao, Michael Jordan, and Zheng-Jun Zha. Rank diminishing in deep neural networks. *Advances in Neural Information Processing Systems*, 35: 33054–33065, 2022.

Ronald A Fisher. On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 222(594-604):309–368, 1922. doi: 10.1098/rsta.1922.0009.

Quentin Garrido, Randall Balestriero, Laurent Najman, and Yann Lecun. Rankme: Assessing the downstream performance of pretrained self-supervised representations by their rank. In *International conference on machine learning*, pp. 10929–10974. PMLR, 2023.

Connall Garrod and Jonathan Keating. The persistence of neural collapse despite low-rank bias. *Advances in Neural Information Processing Systems*, 38:8860–8916, 2026.

Suriya Gunasekar, Blake E Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. Implicit regularization in matrix factorization. *Advances in neural information processing systems*, 30, 2017.

Minyoung Huh, Hossein Mobahi, Richard Zhang, Brian Cheung, Pulkit Agrawal, and Phillip Isola. The low-rank simplicity bias in deep networks. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=bCiNWDmlY2.

Arthur Jacot. Implicit bias of large depth networks: a notion of rank for nonlinear functions. *International Conference on Learning Representations*, 2023.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. *International Conference on Learning Representations*, 2022.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Kenta Oono and Taiji Suzuki. Graph neural networks exponentially lose expressive power for node classification. *International Conference on Learning Representations*, 2020.

Vardan Papyan, X. Y. 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, Sep 2020. ISSN 1091-6490. doi: 10.1073/pnas.2015509117. URL http://dx.doi.org/10.1073/pnas.2015509117.

Jeffrey Pennington, Samuel Schoenholz, and Surya Ganguli. Resurrecting the sigmoid in deep learning through dynamical isometry: theory and practice. *Advances in neural information processing systems*, 30, 2017.

Ben Poole, Subhaneil Lahiri, Maithra Raghu, Jascha Sohl-Dickstein, and Surya Ganguli. Exponential expressivity in deep neural networks through transient chaos. *Advances in neural information processing systems*, 29, 2016.

Noam Razin and Nadav Cohen. Implicit regularization in deep learning may not be explainable by norms. *Advances in neural information processing systems*, 33:21174–21187, 2020.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *2007 15th European Signal Processing Conference*, pp. 606–610, 2007.

Mark Rudelson and Roman Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *Journal of the ACM (JACM)*, 54(4):21–es, 2007.

Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *International Conference on Learning Representations*, 2014.

Samuel S Schoenholz, Justin Gilmer, Surya Ganguli, and Jascha Sohl-Dickstein. Deep information propagation. *International Conference on Learning Representations*, 2017.

Peter Súkeník, Christoph H. Lampert, and Marco Mondelli. Neural collapse vs. low-rank bias: Is deep neural collapse really optimal? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=0jld45XGgJ.

APPENDIX

A TRAJECTORY ARCHETYPES AND OPTIMIZATION OUTCOMES

Table S.1: **Trajectory archetype distribution and optimization success across three random initialization seeds.** Each seed evaluates the same $N = 2{,}268$ configurations, yielding $6{,}804$ fully trained runs in total. Success is defined as $\mathcal{L}_{\text{train}} < 0.01$.

Seed	A1 (%)	A2 (%)	A3 (%)	DRC (%)	A2 Succ. (%)	A3 Succ. (%)	DRC Succ. (%)
42	31.70%	5.42%	21.08%	41.80%	80.49%	32.43%	1.48%
43	33.55%	4.72%	20.50%	41.23%	88.79%	30.97%	1.28%
44	33.99%	5.34%	18.21%	42.46%	81.82%	34.87%	1.56%
Mean ± SD	$33.08 \pm 1.22\%$	$5.16 \pm 0.38\%$	$19.93 \pm 1.52\%$	$41.83 \pm 0.62\%$	$83.70 \pm 4.46\%$	$32.75 \pm 1.97\%$	$1.44 \pm 0.14\%$

Table S.2: **Summary statistics across trajectory archetypes.**

Type	M_{models}	%	Mean Loss	Median Loss	Mean SR
A1	1,746	42.75	0.7127	0.0887	7.8135
A2	187	4.58	**0.0580**	**0.00001**	2.7860
A3	1,045	25.59	0.1846	0.0223	1.1748
DRC	1,106	27.08	$1.05 \times 10^{15\dagger}$	0.3676	1.00

† Diverged during optimization with exploded loss values.

Table S.3: **Success probability conditioned on trajectory archetype.** The probability of successful optimization varies substantially across A1–A3 and DRC.

	$< 10^{-2}$	$< 10^{-3}$
A1	37.80%	35.11%
A2	**83.42%**	**82.89%**
A3	43.73%	40.96%
DRC	1.36%	0.90%

Table S.4: **Architecture class-wise DRC breakdown.** Frozen = DRC runs that did not diverge ($L \leq 100$), this includes runs on low-rank teachers that reach low absolute loss.

Architecture	Total	DRC	DRC Rate	Diverged	Frozen
Deep Linear	252	152	60.32%	15	137
Plain MLP	1,788	780	43.62%	39	741
Residual MLP	2,044	174	8.51%	98	76
Total Synthetic Pool	**4,084**	**1,106**	**27.08%**	**152**	**954**

Table S.5: The 2,268-configuration grid is balanced across initialization schemes (1,134 configurations each per seed). Counts are summed over seeds 42, 43, 44.

Init	A1	A2	A3	DRC	Total
Default (PyTorch Kaiming-uniform; Table S.14)	1,649 (48.5%)	215 (6.3%)	542 (15.9%)	996 (29.3%)	3,402
Low-rank ($r = 1$)	602 (17.7%)	136 (4.0%)	814 (23.9%)	1,850 (54.4%)	3,402
Total	**2,251 (33.1%)**	**351 (5.2%)**	**1,356 (19.9%)**	**2,846 (41.8%)**	**6,804**

B FORMAL HYPOTHESIS TESTING FOR TASK-RANK INVARIANCE

To evaluate whether the target task rank ($r \in \{1, 4, 16, 24, 32, 64\}$) influences representation collapse, we performed statistical tests over all $N = 2{,}268$ synthetic models.

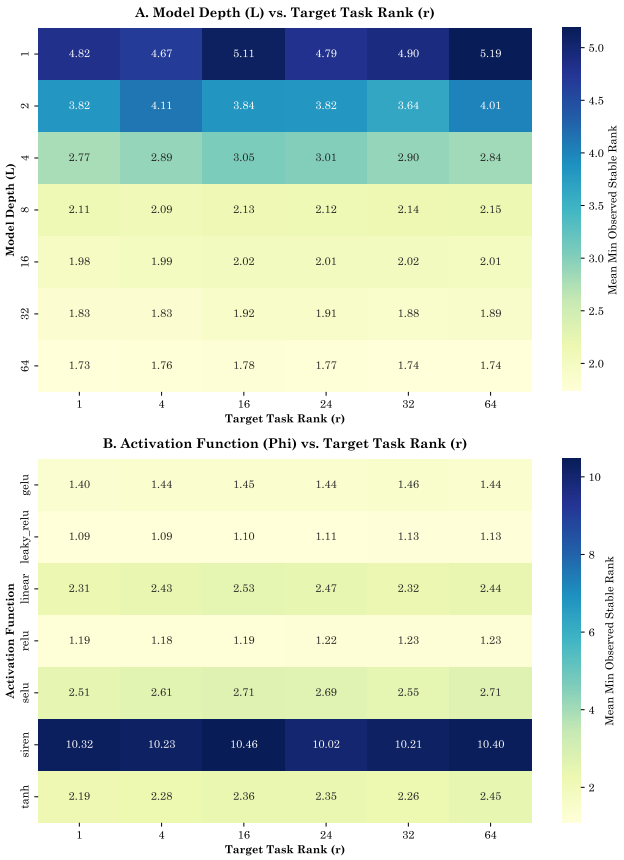


Figure S.1: Mean min-observed representation rank remains uniform across target task ranks $r \in \{1, \ldots, 64\}$, confirming that early-layer rank collapse is governed by architectural depth and activation geometry rather than target task rank.

Table S.6: Statistical Analysis of the Influence of Task Rank on Representation Metrics. SR and ER denote Stable Rank and Effective Rank, respectively.

Target Rank	Runs (N)	Mean SR	Median SR	Mean ER	Median ER
$r = 1$	378	2.7229 ± 3.997	1.0003	8.8891 ± 11.482	1.1900
$r = 4$	378	2.7630 ± 4.010	1.0004	8.9129 ± 11.451	1.2003
$r = 16$	378	2.8345 ± 4.110	1.0003	8.9942 ± 11.449	1.2088
$r = 24$	378	2.7768 ± 3.960	1.0004	9.0070 ± 11.452	1.2312
$r = 32$	378	2.7448 ± 3.966	1.0004	8.9510 ± 11.414	1.1969
$r = 64$	378	2.8307 ± 4.108	1.0008	8.9635 ± 11.419	1.2254

Spearman Rank Correlation. The null hypothesis (H_0) assumes no monotonic relationship between task rank and minimum representation rank. Results indicate no significant correlation for either stable rank ($\rho = 0.0076,\ p = 0.7162$) or effective rank ($\rho = 0.0040,\ p = 0.8499$), failing to reject H_0.

Conclusion. These results show no evidence of a monotonic association between target task rank and minimum representation rank. See Fig. S.1.

The $1,344$ synthetic runs of Table 16 indicate that representational stability depends strongly on the residual variant. Learnable-gate connections retain the most rank (A1 = 78.21%, DRC = 2.14%). Unscaled post-activation variants with unbounded activations (ReLU, Leaky ReLU) at high depth($L \geq 32$) are associated with numerical loss divergence rather than slow rank collapse (Figs. S.2 - S.3).

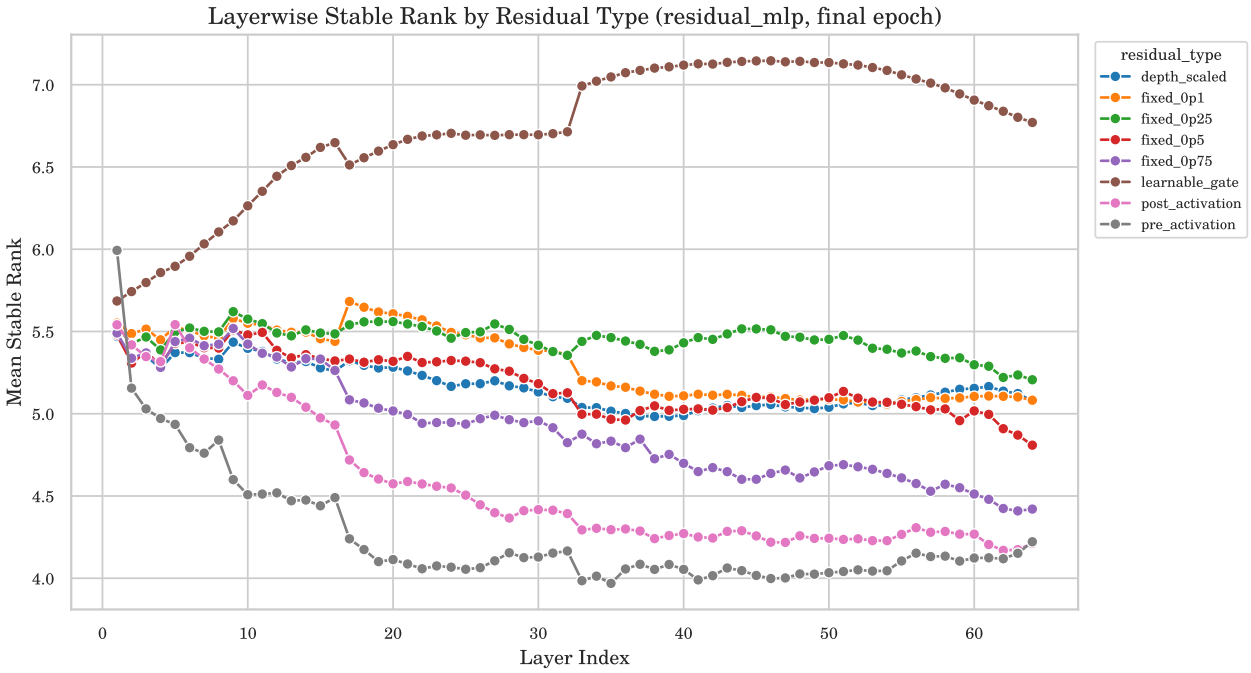


Figure S.2: Observation on Layer-wise Stable rank categorized by different residual implementation styles of Residual MLP network type. Values are calculated at the final epoch so as to find which implementation style would preserve rank to the deepest layer. Values are the mean over Residual-MLP runs that have a layer of the given index.

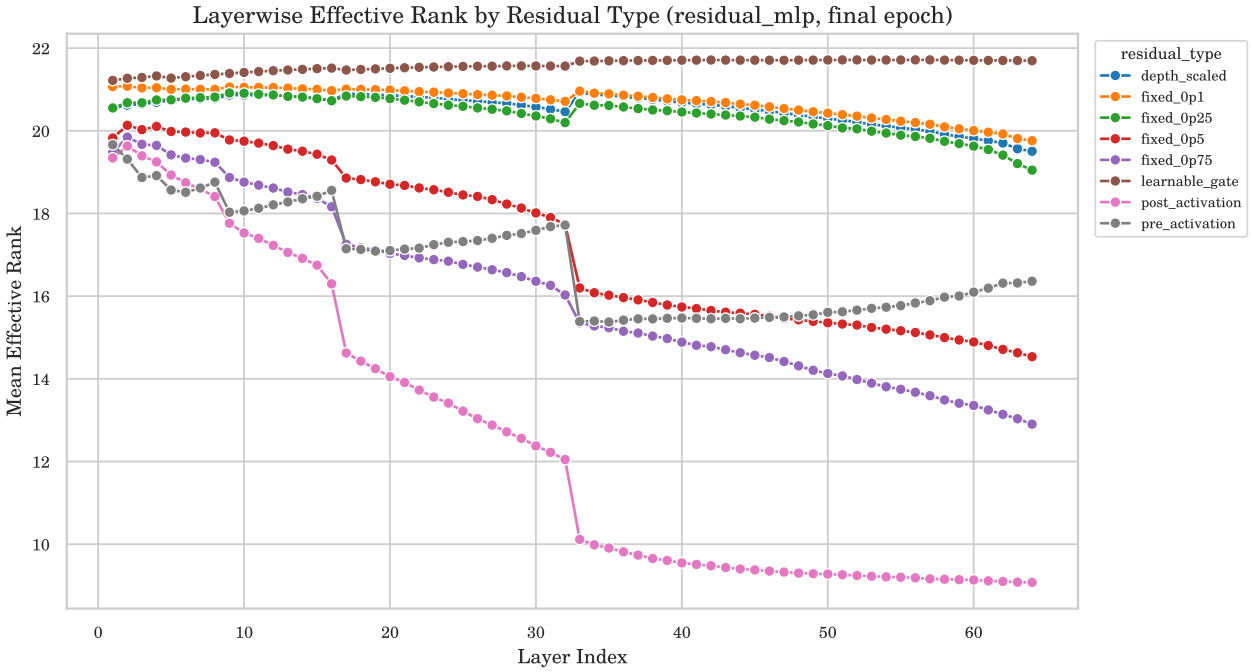


Figure S.3: Observation on Layer-wise Effective rank categorized by different residual implementation styles of Residual MLP network type. Values are calculated at the final epoch so as to find which implementation style would preserve rank to the deepest layer. Values are the mean over Residual-MLP runs that have a layer of the given index.

C Choice of the DRC Thresholds

Justification of Threshold Parameters. The exact quantitative bounds defining DRC ($\text{SR}_{\min} \leq 1.05$, $\text{SR}_{\text{final}} \leq 1.05$, $\text{ER}_{\text{final}} \leq 1.5$) are grounded in both singular value spectral theory and empirical performance phase transitions across $N = 4,390$ models:

1. **Stable Rank Threshold (SR ≤ 1.05)**: $\text{SR} = 1$ exactly when the representation is a single direction ($\sigma_2 = 0$). $\text{SR} \leq 1.05$ means the leading singular value carries at least $1/1.05 \approx 95.24\%$ of the squared Frobenius mass. We use this margin as an empirical tolerance for small residual spectral mass; it is not a floating-point tolerance. Requiring both $\text{SR}_{\min} \leq 1.05$ and $\text{SR}_{\text{final}} \leq 1.05$ excludes runs that recover.

2. **Effective Rank Threshold (ER ≤ 1.5)**: Effective Rank is defined as the exponentiated Shannon entropy of the normalized singular values, where $p_i = \sigma_i / \sum_k \sigma_k$:

$$\text{ER}(A) = \exp\left(-\sum_{i=1}^{R} p_i \ln p_i\right). \tag{2}$$

An upper bound of $ER \leq 1.50$ corresponds to a maximum spectral entropy of $H(p) \leq \ln(1.50) \approx 0.4055$ nats. At this entropy level, the leading singular value σ_1 accounts for more than 85%-95% of the total singular value mass, leaving only a small fraction distributed across the remaining singular values. Consequently, the feature representation becomes effectively confined to a narrow one-dimensional linear corridor, irrespective of the ambient feature dimension.

Emperical Check A rank-one final representation cannot separate more than one class, making chance-level accuracy inevitable for any $k > 1$ classification task. The 192-run CIFAR-10 Plain CNN sweep illustrates this: all runs with $ER_{\text{final}} \leq 1.5$ remain at chance accuracy ($\leq 10.15\%$), while runs with $ER_{\text{final}} > 1.5$ reach up to 91.01% test accuracy. Since the threshold was selected using this same sweep, these results serve as a consistency check rather than an independent validation.

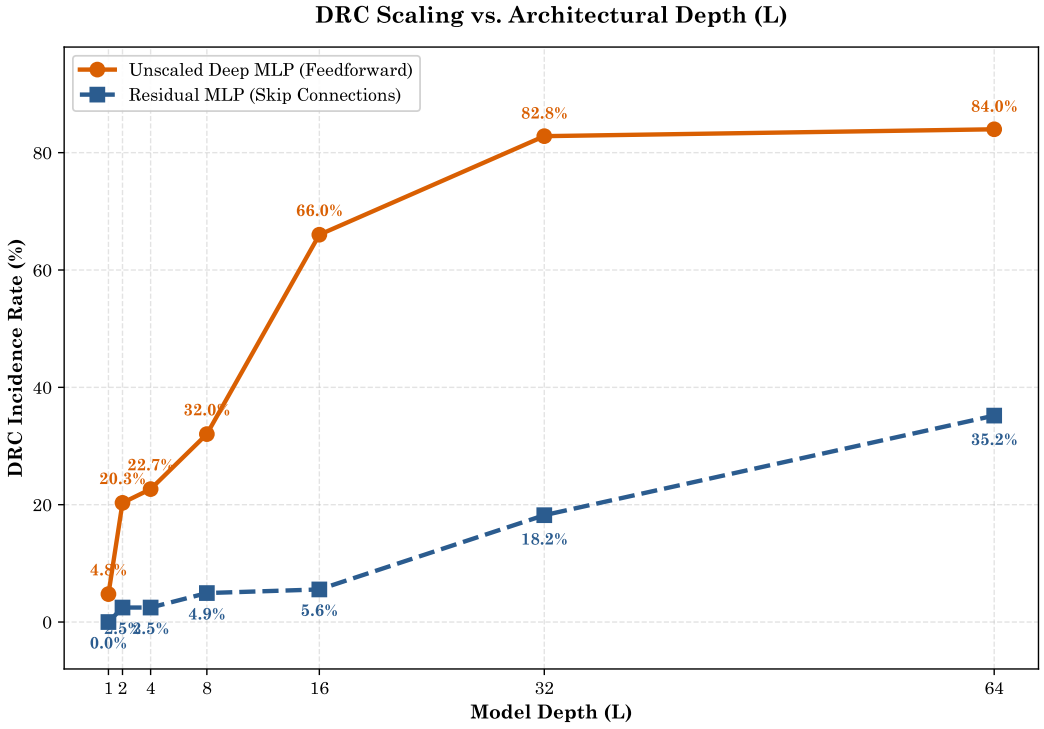


Figure S.4: **DRC incidence versus model depth.** DRC incidence (%) increases nearly monotonically with model depth in both unscaled Deep Plain MLPs (without depth-dependent scaling such as $1/\sqrt{L}$ or ReZero gating) and Residual MLPs.

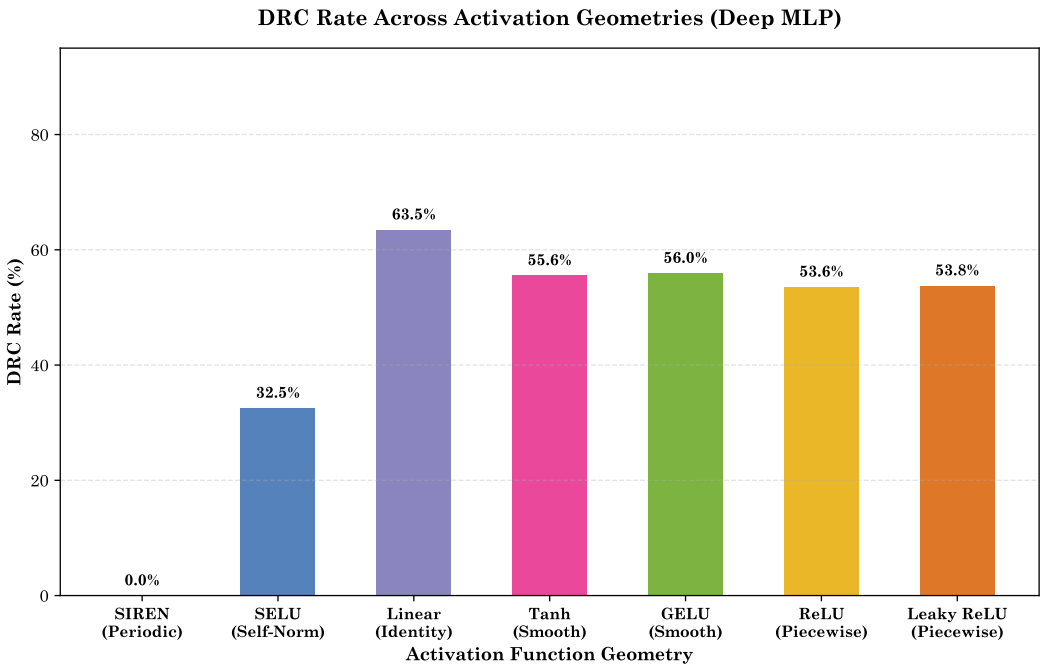


Figure S.5: **Activation-wise susceptibility to DRC.** DRC incidence across activation functions for $N_{\text{models}} = 1,764$ models. SIREN shows 0%DRC yet has the lowest success (Table 3)

Table S.7: **Milestone statistics for a benign low-rank trajectory.** The network maintains low stable rank (SR ≈ 1.05–1.19) while preserving effective rank (ER ≥ 10.4), enabling it to achieve 90.35% CIFAR-10 test accuracy.

Epoch	Acc. (%)	Min SR	Mean SR	Min ER	Mean ER	Loss
1	9.92	1.191	2.245	13.54	165.85	2.515
5	12.45	1.102	1.813	13.97	143.40	2.336
10	47.62	1.213	1.865	14.19	88.25	1.429
25	76.97	1.163	2.136	18.50	64.85	0.705
50	83.49	1.129	2.654	19.77	53.43	0.520
100	86.54	1.109	2.637	18.27	44.83	0.511
150	89.82	1.082	2.749	16.16	40.25	0.386
200	87.47	1.072	3.201	13.95	37.98	0.439
250	90.63	1.058	3.138	11.61	35.41	0.350
300	90.35	1.051	3.040	10.39	33.11	0.362

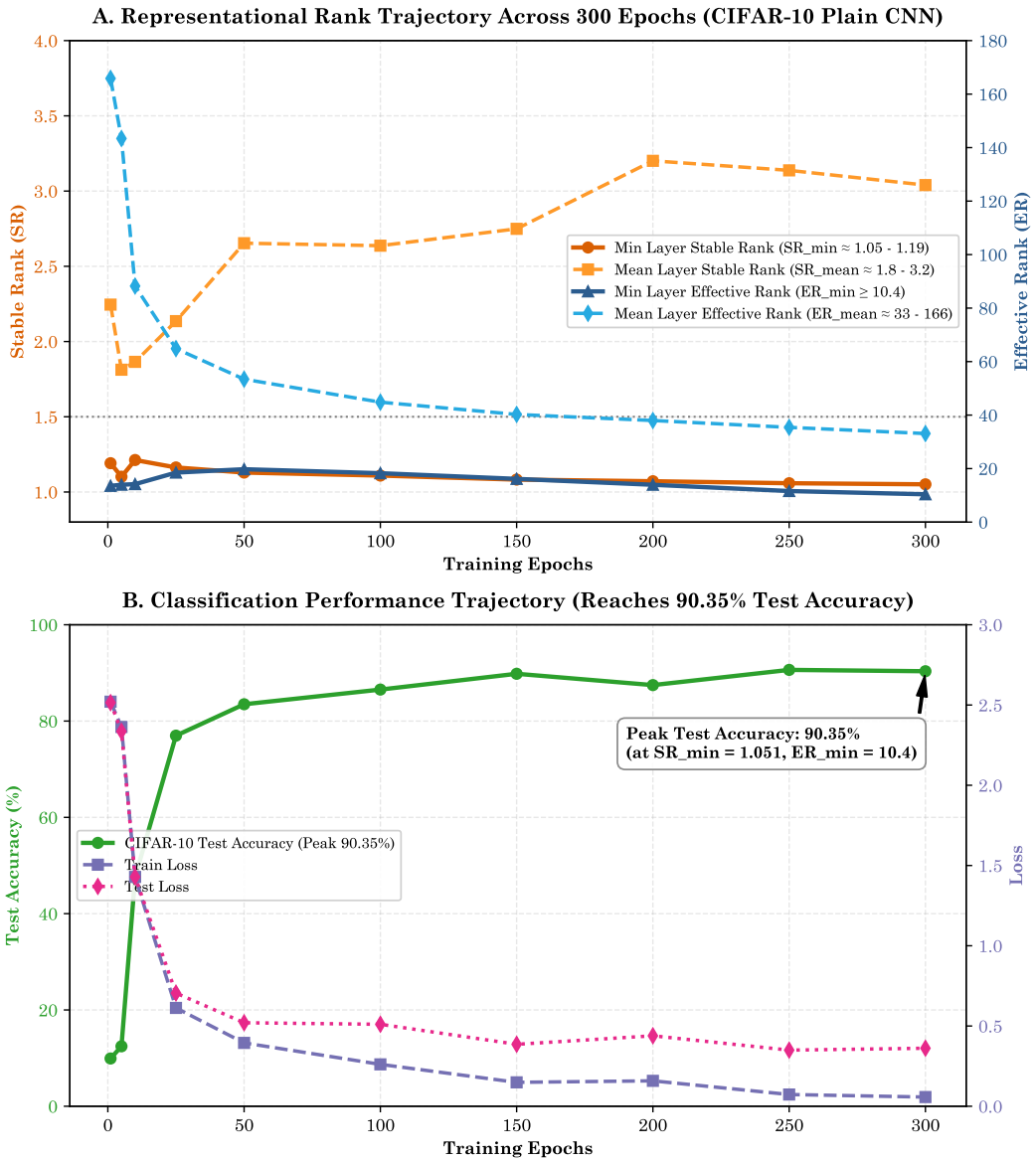


Figure S.6: **Low stable rank does not imply DRC.** Despite $\mathrm{SR}_{\min} \approx 1.05$, preserved effective rank ($\mathrm{ER}_{\min} \geq 10.4$) enables the model to reach 90.35% CIFAR-10 test accuracy.

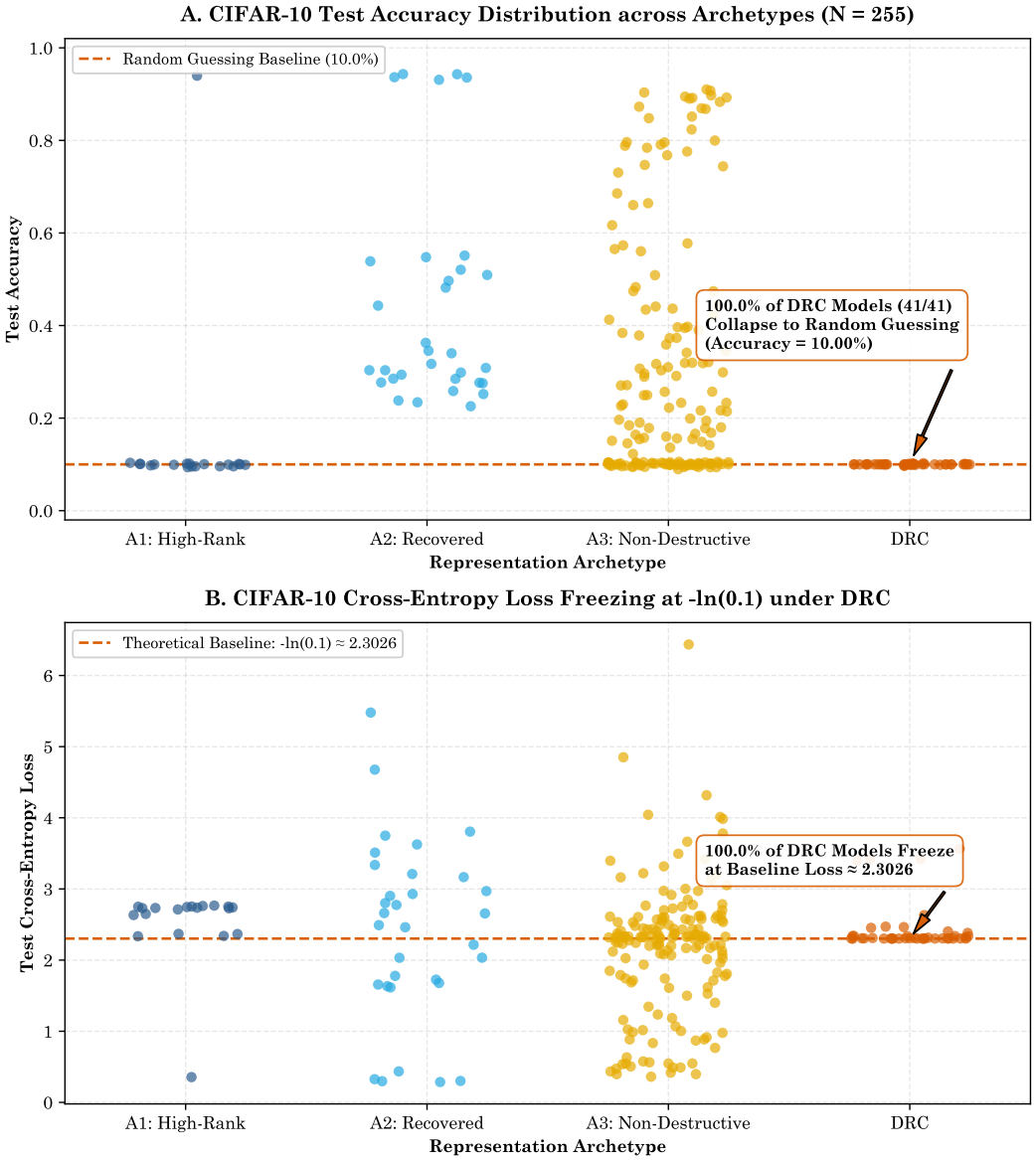


Figure S.7: **Classification performance across A1–A3 and DRC.** A1–A3 achieve high test accuracy, whereas all DRC models remain at random-guessing accuracy ($\approx$ 10%) with cross-entropy fixed at the theoretical baseline ($-\ln(0.1) \approx 2.3026$). ViT runs are analyzed separately in Appendix G

Table S.8: **Common hyperparameter specifications across synthetic and vision experiment sweeps.**

Hyperparameter	**Synthetic Sweeps** ($N = 4{,}084$)	**CIFAR-10 Plain CNN sweep** ($N = 192$)
Random Seed	42	42
Learning Rate (η)	0.001	0.001
Weight Decay (λ)	0.0	0.0001
Batch Size (B)	Full-batch (32)	128
Optimizers	Adam, Momentum SGD, SGD	Adam, SGD
Loss Criterion	Mean Squared Error (MSE)	Cross-Entropy Loss
Input/Output Dimension	$64 \to 64$	$3 \times 32 \times 32 \to 10$

D Statistical Background

This section briefly summarizes the statistical measures used throughout the paper.

D.1 Odds Ratio (OR)

The *Odds Ratio (OR)* quantifies the strength of association between two groups. It is defined as the ratio of their odds of success:

$$\mathrm{OR} = \frac{\dfrac{p_A}{1-p_A}}{\dfrac{p_B}{1-p_B}}, \tag{3}$$

where p_A and p_B denote the probabilities of success for Groups A and B, respectively.
Interpretation:

- $\mathrm{OR} = 1$: no association.
- $\mathrm{OR} > 1$: Group A has higher odds of success.
- $\mathrm{OR} < 1$: Group A has lower odds of success.

D.2 Adjusted Odds Ratio (AOR)

When multiple variables may jointly influence the outcome, a multivariate logistic regression model is used:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k. \tag{4}$$

The adjusted odds ratio for predictor X_i is

$$\mathrm{AOR}_i = \exp(\beta_i), \tag{5}$$

which represents the independent effect of X_i while controlling for all other predictors.

D.3 Fisher's Exact Test

Fisher's Exact Test evaluates the association between two categorical variables in a 2×2 contingency table. Unlike the χ^2 test, it computes the exact probability of observing the contingency table under the null hypothesis:

$$\text{P(table)} = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{a!\,b!\,c!\,d!\,n!}, \tag{6}$$

where $n = a + b + c + d$.
The two-sided p-value sums P over all tables with the same margins that are no more probable than the observed table

D.4 Confidence Intervals and p-values

A 95% confidence interval (CI) provides a plausible range for the true population effect size. For odds ratios, statistical significance is typically inferred when the 95% CI does not include 1. The confidence interval is computed as

$$\exp(\beta_i \pm 1.96\,\mathrm{SE}(\beta_i)). \tag{7}$$

The p-value denotes the probability of observing an effect at least as extreme as the measured one under the null hypothesis. Conventionally,

- $p < 0.05$: statistically significant,
- $p < 0.001$: highly statistically significant.

D.5 Receiver Operating Characteristic (ROC) Curve and AUC

A Receiver Operating Characteristic (ROC) curve plots the True Positive Rate (Sensitivity) against the False Positive Rate (1–Specificity) across all classification thresholds.
The Area Under the ROC Curve (AUC) summarizes overall classification performance:

- $\mathrm{AUC} = 0.5$: random prediction.
- $\mathrm{AUC} = 1.0$: perfect prediction.

E Early-Trajectory Prediction of Optimization Outcomes

To examine whether representation dynamics contain information about optimization outcomes before convergence, we classified models using trajectory information available at early training

milestones of 10%, 25%, and 50% of training. For synthetic MLPs, these correspond to epochs 100, 250, and 500 of a 1,000-epoch run. Models were assigned to an early trajectory category based on the minimum stable rank observed up to each milestone and its subsequent change: **Never Collapsed** if $SR_t > 1.50$ throughout the observed interval, **Early Recovery** if the representation had collapsed but subsequently expanded, and **Early Stuck** if it remained near the collapse bottleneck. The classification uses only information available up to the respective milestone and does not use final loss or final representations.

Across 2,268 synthetic MLP runs, the early trajectory categories differ in their eventual optimization outcomes. At 25% of training, 618 models were classified as Early Recovery, with a 35.8% success rate, compared with 4.9% among the 930 Early Stuck models. This corresponds to a relative risk of 7.16× and an odds ratio of 10.60. Similar differences are observed at 10% and 50%, with odds ratios of 6.33 and 12.11, respectively. These comparisons are unadjusted for configuration (activation, optimizer, depth, initialization) and should be read as descriptive.

Overall, these results suggest that representation trajectories contain information that is related to eventual optimization outcomes before convergence. In particular, models that remain near the rank-collapse bottleneck early in training tend to have lower subsequent success rates, whereas models showing early recovery tend to have better outcomes. This result is consistent with early trajectory carrying forecasting information, we do not yet test whether it can be turned into a reliable automated early-stopping criterion, which we leave to future work.

E.1 Cross Validated Association

Table S.9: **5-fold stratified cross-validation performance** ($M_{\text{models}} = 4{,}084$)**.** SD=Standard deviation

Model	Mean AUC	SD	95% CI	Acc.
[1]Static Final-Rank Endpoint	0.7743	0.0193	[0.7574, 0.7912]	69.20%
[2]Hyperparameter Controls Only	0.9459	0.0073	[0.9395, 0.9523]	90.77%
[3]Full Trajectory + Controls	**0.9660**	**0.0048**	**[0.9618, 0.9702]**	**92.58%**

[1]Uses only final-epoch rank snapshots (SR_{final}, ER_{final}). [2]Uses setup features only (activation, architecture, depth, optimizer, rank). [3]Combines temporal trajectory archetypes with setup features.

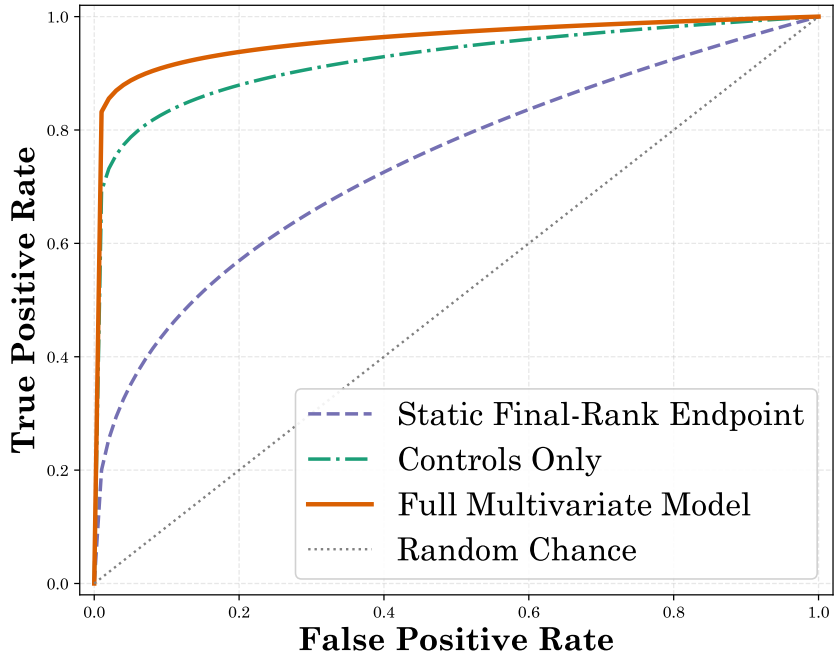


Figure S.8: 5-Fold Cross-Validation ROC Curve Comparison

F Sensitivity to Trajectory Classification Thresholds

This is a robustness check of the early-milestone forecasting result in Appendix E. To examine how sensitive the trajectory–outcome relationship is to the chosen numerical boundaries, we repeated

Table S.10: Depth-Wise Representation & Performance Scaling ($N = 48$ Models, 200 Epochs)

Architecture	Depth (L)	N	Train Acc (%)	Test Acc (%)	Deepest Block SR	Deepest Block ER	Penult SR	Catastrophic DRC
Standard ViT *(Pre-LN + Residual)*	3	8	87.0 ± 8.7	76.2 ± 3.6	6.50 ± 1.85	72.7 ± 23.8	5.19 ± 0.49	**0 / 8 (0.0%)**
Standard ViT *(Pre-LN + Residual)*	6	8	83.2 ± 17.5	72.3 ± 10.6	6.27 ± 2.00	71.2 ± 22.8	5.08 ± 1.09	**0 / 8 (0.0%)**
Standard ViT *(Pre-LN + Residual)*	9	8	91.9 ± 8.6	77.2 ± 3.6	7.25 ± 1.35	75.1 ± 25.4	5.79 ± 0.78	**0 / 8 (0.0%)**
No-Residual ViT *(Pre-LN Only)*	3	8	74.3 ± 6.8	68.0 ± 4.8	3.53 ± 0.51	29.7 ± 26.5	3.58 ± 0.59	**0 / 8 (0.0%)**
No-Residual ViT *(Pre-LN Only)*	6	8	66.6 ± 8.8	62.8 ± 7.8	2.94 ± 0.62	22.9 ± 19.7	3.29 ± 0.27	**0 / 8 (0.0%)**
No-Residual ViT *(Pre-LN Only)*	9	8	52.8 ± 27.0	49.2 ± 24.6	2.46 ± 1.04	13.5 ± 11.6	2.62 ± 1.07	**2 / 8 (25.0%)**

Table S.11: Layer-by-Layer Metrics across Depth at $L = 9$ ($N = 16$ Runs, Epoch 200)

Layer	Standard ViT SR	Standard ViT ER	No-Residual ViT SR	No-Residual ViT ER
Block 1	5.95 ± 1.19	57.7 ± 13.8	3.49 ± 2.39	39.7 ± 27.1
Block 2	7.08 ± 1.40	68.8 ± 20.2	3.00 ± 2.35	29.4 ± 31.3
Block 3	7.40 ± 1.92	73.3 ± 22.8	2.66 ± 1.90	21.6 ± 26.8
Block 4	8.10 ± 1.68	75.4 ± 23.4	2.55 ± 1.65	15.7 ± 18.8
Block 5	8.34 ± 1.94	76.6 ± 23.8	2.68 ± 1.75	12.4 ± 13.9
Block 6	8.82 ± 2.43	77.2 ± 23.9	2.78 ± 1.81	11.5 ± 12.2
Block 7	8.54 ± 1.99	76.9 ± 23.2	2.65 ± 1.70	11.3 ± 11.9
Block 8	8.69 ± 1.67	76.8 ± 23.3	2.67 ± 1.75	11.7 ± 11.9
Block 9 *(Deepest)*	7.25 ± 1.27	75.1 ± 23.7	2.46 ± 0.97	13.5 ± 10.9
Penultimate Layer	5.79 ± 0.73	59.2 ± 20.4	2.62 ± 1.00	13.4 ± 10.7

the archetype classification at the 25% training milestone $(epoch250)$ across a grid of collapse and recovery thresholds. The collapse threshold τ_{collapse} was varied over $\{1.05, 1.10, 1.15, 1.20, 1.50\}$, and the recovery threshold $\Delta\tau_{\text{rec}}$ over $\{0.05, 0.10, 0.15, 0.20\}$, giving 20 threshold combinations. Across this grid, the relationship between trajectory type and optimization outcome is similar across the tested settings. Recovery trajectories have higher success rates than early-stuck trajectories, with odds ratios ranging from 7.35 to 15.71 and relative risks from 5.25 to 10.69. Recovery success rates range from 31.4% to 37.7%, compared with 3.1% to 6.8% for the early-stuck group. Changing the thresholds therefore changes the number of models assigned to each archetype, but the separation between recovery and persistent low-rank trajectories remains present across the tested boundaries.

G Extension to V ision Transformers

To examine whether the representation trajectory framework also extends beyond feedforward and convolutional architectures, we evaluated 48 CIFAR-10 Vision Transformer runs using both standard residual and no-residual configurations at depths $L \in \{3, 6, 9\}$. We tracked Stable Rank (SR) and Effective Rank (ER) throughout training and classified the early trajectories using only the information available up to the corresponding training milestone.

Standard residual ViTs maintain substantially higher representation rank across depth, while no-residual ViTs show a progressive contraction in rank. At depth 9, the no-residual models have substantially lower effective rank than their residual counterparts, with 2 of the 8 evaluated runs satisfying the DRC criterion. Both of these runs reach approximately rank-one representations and achieve chance-level accuracy on CIFAR-10. See Table S.10.

The layer-wise analysis further shows that this difference builds up with depth. In the evaluated depth-9 models, the effective rank of the no-residual configuration decreases from approximately 39.7 at the first block to 11.3 at the most contracted layer, while the corresponding residual models maintain substantially higher effective rank. This provides an empirical example of depth-dependent representation contraction occurring when skip connections are absent. See Table S.11.

This replicates, in an independent architecture family, the early-forecasting direction observed in the synthetic MLP sweeps (Appendix E): at 25% of training, 11 of 15 Early Recovery models eventually succeed, compared with 0 of 2 Early Stuck/DRC models. The stuck-group count is small ($n = 2$), so the magnitude should be treated as indicative rather than precise; the consistency in direction with the synthetic results is the main takeaway. See Table S.12.

H Dataset Configuration

H.1 Synthetic Low-Rank Regression:

The data description in detail has been already mentioned in the main section of the paper.

Table S.12: Early Trajectory Archetype Forecasting in Vision Transformers

Training Milestone	Arch 1: Never Collapsed (Success %)	Arch 2: Early Recovery (Success %)	Arch 3: Early Stuck (Success %)	Odds Ratio (Arch 2 vs 3)	Mean Final Accuracy (A2 vs A3)
10% of Training *(Epoch 20)*	31 (90.3%)	**14** (78.6%)	3 (0.0%)	23.00×	64.4% vs 25.9%
25% of Training *(Epoch 50)*	31 (90.3%)	**15** (73.3%)	2 (0.0%)	12.78×	64.0% **vs** 10.0%
50% of Training *(Epoch 100)*	31 (90.3%)	**15** (73.3%)	2 (0.0%)	12.78×	64.0% **vs** 10.0%

H.2 CIFAR- 10 IMAGE CLASSIFICATION:

Dataset:$50,000$ training and $10,000$ testing images across 10 classes are used.
Preprocessing: Normalization using CIFAR-10 means and standard deviations. Training images are augmented with random crops and random horizontal flips.

I EXPERIMENTAL SWEEPS CENSUS AND ARCHITECTURAL SPECIFICATIONS

I.1 ARCHITECTURAL SPECIFICATIONS

Optimization algorithms used - Table S.13
Initialization Schemes used - Table S.14
Activation Functions used - Table S.15
Residual Connection Variants used - Table S.16
Description of Plain MLP and DeepLinear MLP architectures used - Table S.17
Description of Residual MLP used. - Table S.18
Description of Plain CNN used. - Table S.19
Description of ResNet-18 and ResNet-50 used - Table S.20
Description of ViT used - Table S.21

Table S.13: Optimization algorithms used in the experimental study. All optimizers use learning rate $\eta = 10^{-3}$ and weight decay $\lambda = 0$ (synthetic), 10^{-4} (CIFAR-10 CNN)

Optimizer	**Description**	**Default Hyperparameters**
SGD	Plain gradient descent	—
SGD + Momentum	Momentum-based GD	$\beta = 0.9$
Adam	Adaptive moment estimation	$\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$

Table S.14: Initialization schemes evaluated.

Scheme	**Initialization**	**Key Parameter**
Default	PyTorch Kaiming Uniform	–
Low-Rank	$W = \dfrac{AB}{\sqrt{r d_{\text{in}}}},\ A, B \sim \mathcal{N}(0,1)$	Rank r ($r = 1$)

Table S.15: Activation functions used in the experiments.

Activation	**Definition**
Linear	$\Phi(x) = x$
ReLU	$\Phi(x) = \max(0, x)$
Leaky ReLU	$\Phi(x) = \max(0.01x, x)$
Tanh	$\Phi(x) = \tanh(x)$
GELU	Standard GELU approximation
SELU	Standard SELU activation
SIREN	$\Phi(x) = \sin(\omega_0 x),\ \omega_0 = 30$

Table S.16: Residual connection variants. Depth scaled variant defined for $L \geq 2$.

Variant	**Residual Mapping**
Post-activation	$h^{(l)} = h^{(l-1)} + \Phi(W^{(l)} h^{(l-1)})$
Pre-activation	$h^{(l)} = h^{(l-1)} + W^{(l)} \Phi(h^{(l-1)})$
Depth-scaled	$h^{(l)} = h^{(l-1)} + \frac{1}{L-1} \Phi(\cdot)$
Fixed-scale	$h^{(l)} = h^{(l-1)} + \alpha \Phi(\cdot)$
Learnable gate (ReZero-style)	$h^{(l)} = h^{(l-1)} + \alpha_l \Phi(\cdot)$

Table S.18: Description of Residual MLP used.

ResidualMLP
Input: Flattened Vector
Linear ($d_{\text{in}} \to H$)
$(L-1)\times$ Residual Block ($H \to H$)
Linear ($H \to d_{\text{out}}$)
Residual connections begin from Block 2.

Table S.19: Description of Plain CNN used.

PlainCNN
Input: RGB Image (32×32)
Conv 3×3, BN, Activation
$(L-1)\times$ Conv 3×3, BN, Activation
MaxPool at regular intervals (see released code)
Global Average Pooling
Linear Classifier

Table S.17: Description of Plain MLP and DeepLinear MLP architectures used

Plain MLP / DeepLinear
Input: Flattened Vector
Linear ($d_{\text{in}} \to H$)
$(L-1)\times$ Linear ($H \to H$)
Linear ($H \to d_{\text{out}}$)
DeepLinear uses identity activation throughout.
Note: L counts hidden layers, the network has $L+1$ weight matrices.

I.2 Experimental Sweeps Census

Across the entire benchmark library, we evaluated $4{,}084$ synthetic models and 306 real-world CIFAR-10 vision models (Total = 4,390 trained deep learning models) RTX 2080 Ti GPUs and PyTorch have been used for the experimentation purposes.

I.2.1 Synthetic Regression Sweeps Library $M_{\text{models}} = 4{,}084$ Models

Target Task - Low-Rank Regression
Description of Synthetic Full Grid Sweep -Table S.22
Description of Synthetic Recovery Grid Sweep - Table S.23
Description of Synthetic Residual Runs 1 - Table S.24
Description of Synthetic Residual Runs 2 extended - Table S.25

Synthetic data Full grid Sweep This experiment is conducted on a full grid of multiple architectures, optimizers, initialization schemes, task ranks, depths etc. To get a full generalized view of how a network's representation's dynamic changes as per different configurations. **Total Runs -** $2,268$, **Epochs -** 1000 , Table S.22

Table S.20: Description of ResNet-18 and ResNet-50 used

ResNet-18 / ResNet-50
Input: RGB Image (32×32)
3×3 Conv Stem, BN, ReLU
Stage 1: Residual Blocks (64)
Stage 2: Residual Blocks (128, stride 2)
Stage 3: Residual Blocks (256, stride 2)
Stage 4: Residual Blocks (512, stride 2)
Global Average Pooling
Linear Classifier
BasicBlock (ResNet-18), Bottleneck (ResNet-50)

Table S.21: Description of Vision Transformer (ViT) used.

Vision Transformer (ViT)
Input: RGB Image (32×32)
Patch Embedding: Conv 4×4, Stride 4, Dim d (64 Patches)
Prepend [CLS] Token ($1 \times d$) + Positional Embedding ($65 \times d$)
$L\times$ Pre-LN Transformer Blocks ($L \in \{3, 6, 9\}$):
Multi-Head Self-Attention ($h \in \{2, 4\}$, $d_k = 32$) $\pm$ Residual
Transformer MLP ($d \to 2d \to d$, Activation) $\pm$ Residual
Final LayerNorm on [CLS] Token (Penultimate Representation)
Linear Classifier ($d \to 10$)

Table S.23: Synthetic Recovery grid sweep.

Setting	**Value**
Architecture	Plain MLP
Runs	24
Activations	SIREN, Leaky ReLU
Depth	$\{2, 4, 8, 16, 32, 64\}$
Task Rank	24
Optimizers	SGD, Adam
Initialization	Low-Rank

Table S.22: Synthetic Full Grid Sweep

Common Settings	
Depth	$\{1, 2, 4, 8, 16, 32, 64\}$
Task Rank	$\{1, 4, 16, 24, 32, 64\}$
Optimizer	SGD, Momentum SGD, Adam
Initialization	Default, Low-Rank
Runs	2268 (total)
Architecture	**Configuration**
Deep Linear (252 runs)	Activation: Linear
Deep Residual MLP (252 runs)	Activation: ReLU Residual: Pre-activation
Plain MLP (1764 runs)	Activations: Linear, ReLU, Tanh, Leaky ReLU, SELU, GELU, SIREN

Synthetic Recovery grid sweep. This particular configuration aims to check whether a longer training time has any effect on recovery or no. Task rank is kept fixed because task rank has no detectable effect on representation dynamics. **Epochs -** $10,000$. Table S.23

Synthetic Residual Runs 1. Experiment to check the effect of different residual implementation strategies on representation dynamics. **Epochs -** 1000. Table S.24

Table S.24: Synthetic Residual Runs Sweep

Setting	**Value**
Architecture	Residual MLP
Runs	448
Depth	$\{1, 2, 4, 8, 16, 32, 64\}$
Activations	SIREN, Leaky ReLU
Residual Types	Post-activation, Depth-scaled, Fixed-0.5, Learnable Gate
Task Ranks	$\{1, 64\}$
Optimizers	SGD, Adam
Initialization	Default, Low-Rank

Synthetic Residual Runs 2. An extended configuration runs of Synthetic Residual Runs 1. **Epochs -** 1000. Table S.25

Table S.25: Synthetic Residual sweep extended.

Setting	Value
Architecture	Residual MLP
Runs	1344
Depth	$\{2, 4, 8, 16, 32, 64\}$
Task Rank	24
Activations	Linear, ReLU, Tanh, Leaky ReLU, SELU, GELU, SIREN
Residual Types	Post-activation, Pre-activation, Depth-scaled, Fixed-0.1, Fixed-0.25, Fixed-0.5, Fixed-0.75, Learnable Gate
Optimizers	Adam, Momentum SGD
Initialization	Default, Low-Rank

I.2.2 Real World Dataset Sweep

Description of CNN, Plain MLP, ResMLP sweep on CIFAR10 - Table S.26
Description of CNN only on CIFAR10 sweep - Table S.27
Description of ResNet18 , ResNet50 on CIFAR10 sweep - Table S.28
Description of ViT sweep - Table S.29

CIFAR10 CNN, MLP, ResMLP sweep. This sweep's aim is to observe how representation dynamics appear across models with different settings on Real world dataset such as CIFAR10. This also checks whether effect of width of a layer and input data sample size. **Total Runs -** 60, **Epochs -** 200

Table S.26: CIFAR10 CNN, Plain MLPs, Residual MLPs sweep

Setting	Value
Architectures	MLP, Residual MLP, CNN
Expansion Factor	$\{1, 2, 4, 8\}$
Samples/Class	Full, 20, 50, 100, 200

CNN only CIFAR10 sweep. This experiment's aim is to observe how representation dynamics work in Plain CNN configurations. **Total Runs -** 192, **Epochs -** 300.

Table S.27: CNN only on CIFAR10 sweep

Setting	Value
Architecture	Plain CNN
Depth	$\{8, 32\}$
Activations	Leaky ReLU, SIREN
Expansion Factor	$\{1, 2, 4\}$
Optimizers	SGD, Adam
Initialization	Default, Low-Rank
Samples/Class	20, 100, 1000, Full

ResNet CIFAR10 sweep. The motivation of this experiment is to observe how representation dynamics operate in standard ResNet-18 and ResNet-50 with 3 different residual connection implementation styles. **Total Runs -** 6, **Epochs**-2500.

Table S.28: ResNet CIFAR10 sweeps

Setting	Value
Architecture	ResNet
Models	ResNet-18, ResNet-50
Residual Skip Modes	On: $x + F(x)$ Off: $F(x)$ Scaled: $x + \frac{1}{N_{blocks}} F(x)$

ViT CIFAR10 sweep. The motivation of this experiment is to observe how representation dynamics operate in Vision Transformers across 3 depths and 2 widths with standard and non-residual connection styles. **Total Runs - 48 , Epochs - 200**

Table S.29: Factorial Grid Sweep Configurations for Vision Transformers ($N = 48$ Models).

Hyperparameter	**Values / Levels**	**Count**
Residual Mode	Standard (with residuals), No-Residual (without residuals)	2
Depth (L)	3, 6, 9 blocks	3
Embedding Dim (d)	64 ($h = 2, d_{\text{mlp}} = 128$), 128 ($h = 4, d_{\text{mlp}} = 256$)	2
Activation (Φ)	GELU, ReLU	2
Optimizer	Adam ($\eta = 10^{-3}$), Momentum SGD ($\eta = 10^{-2}, \beta = 0.9$)	2
Total Configurations	$2 \times 3 \times 2 \times 2 \times 2$	**48**