# SAME FIT, DIFFERENT DAMAGE: LEARNING ORDER AND FUNCTIONAL DEPENDENCE IN NOISY-LABEL MEMORIZATION

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

## ABSTRACT

Deep neural networks can eventually fit nearly all incorrect labels, yet this memorization can cause very different levels of semantic damage. We show that the difference is closely related to when individual errors are learned and how their final wrong predictions are functionally realized. To study this phenomenon, we introduce relational consistency, which describes how strongly an incorrect target is supported by surrounding training examples. Using a controlled intervention that preserves the corrupted examples and class transition statistics while changing only the assignment of wrong targets, we find that highly compatible errors are learned 27.64 epochs earlier on average than random matched errors under strong augmentation. Mid-training gradient alignment further predicts this ordering. Importantly, strong augmentation does not prevent eventual memorization. When both models have fitted 98% of the incorrect labels, the strongly augmented model still achieves 23.1 percentage points higher linear-probe accuracy for the true labels. Additional controls show that errors learned later preserve stronger true-label structure at comparable levels of noise fitting. Suffix rollback experiments further reveal that these later errors depend more strongly on deep network changes formed around their learning events, and removing these changes often restores the true prediction. These patterns persist across datasets, architectures, noise rates, and human annotation noise. Our results reveal that near-identical levels of noise fitting can coexist with substantially different semantic costs, systematically linked to learning order and the functional realization of individual errors.

## 1 INTRODUCTION

Deep neural networks can eventually fit nearly all incorrect labels even under substantial noisy supervision. Classical and modern noisy-label methods address this difficulty through corrected losses, bootstrapping objectives, curriculum-style sample selection, and mixture-based noise modeling (Frénay & Verleysen, 2014; Natarajan et al., 2013; Reed et al., 2014; Patrini et al., 2017; Jiang et al., 2018; Arazo et al., 2019; Song et al., 2022). Yet similar levels of noise fitting do not necessarily cause the same degree of semantic damage. Prior work shows that data augmentation can preserve useful representations even when models ultimately fit large amounts of, or even all, random labels, and augmentation has become a central tool for controlling invariance and robustness in image models (Zhang et al., 2018; Shorten & Khoshgoftaar, 2019; Yun et al., 2019; Cubuk et al., 2020; Hendrycks et al., 2020). At the same time, memorization itself can reflect structured long-tail aspects of the training distribution rather than only pathological overfitting (Feldman & Zhang, 2020). As shown in Figure 1, models trained without augmentation and with strong augmentation both approach

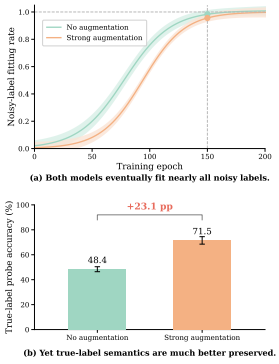



Figure 1: **Strong augmentation reduces semantic damage.** At matched 98% noise fit, it preserves much higher true-label structure.



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.

complete noisy-label fitting, while the latter retains substantially better true-label representations. Thus, how much noise is ultimately memorized is insufficient to characterize the true cost of noisy-label memorization. This raises a more fundamental question: when incorrect labels are eventually memorized, why do different errors incur substantially different representational costs?

We find that this difference is closely associated with two related factors: the order in which erroneous examples are learned and how their final wrong predictions are functionally realized. To study learning order, we introduce relational consistency, which describes how strongly an incorrect target is supported by surrounding training relations. We further construct a strictly controlled target-pairing intervention: the corrupted examples and all source–target class-transition statistics are fixed, while only the pairing between individual examples and incorrect targets is changed. This isolates sample–target compatibility from the noise rate and global class-transition structure.

Under this matched setting, high-compatibility errors under strong augmentation are stably learned 27.64 epochs earlier on average than matched-random errors. Mid-training gradient alignment prospectively predicts subsequent learning order among errors that remain unlearned, revealing a systematic relationship between shared gradient support and when incorrect labels are acquired. Augmentation-switch experiments show that this ordering difference is established mainly during early-to-middle training rather than arising from a simple delay in late-stage noise fitting.

This effect also cannot be explained simply by strong augmentation delaying noise memorization. We therefore compare models at matched noise-fitting events rather than matched epochs. When both conditions have fitted 98% of the incorrect labels, the noisy supervision is already almost equally decodable, yet the strongly augmented model retains 23.1 percentage points higher deep-layer true-label linear-probe accuracy (71.5% vs. 48.4%). Thus, nearly identical noise-fitting levels can correspond to substantially different levels of true-semantic preservation.

To further understand how this difference is realized in the final model, we perform event-aligned suffix rollback, rewinding only the deep suffix around each erroneous example's learning event while keeping the final shallow prefix fixed. Late-learned errors exhibit stronger functional suffix dependence; under strong augmentation, most wrong predictions erased by rollback revert directly to their true classes. This further links learning order to the functional realization and semantic cost of final wrong predictions.

Our main contributions are:

- We reveal systematic learning-order differences among incorrect labels and use a transition-matched target-pairing intervention to isolate the role of sample–wrong-target compatibility while keeping the corrupted examples and all source–target class-transition statistics fixed.
- We show that strong augmentation primarily amplifies these learning-order differences during early-to-middle training, while mid-training gradient alignment provides a localized optimization signature of subsequent learning order.
- Through event-aligned representation analysis and sample-level suffix rollback, we establish a systematic link between learning order, functional suffix dependence, and semantic preservation, providing a sample-level account of how near-complete noise memorization can coexist with substantially different representational costs.

## 2 Related Work

**Noisy-label memorization and learning dynamics.** Overparameterized neural networks can fit arbitrary labels, motivating work on memorization, generalization, and noisy supervision (Bengio et al., 2007; Hinton et al., 2006; Goodfellow et al., 2016; Zhang et al., 2016; Arpit et al., 2017). Examples also exhibit stable forgetting and difficulty patterns (Toneva et al., 2018), and noisy-label training often learns semantic structure before entering a noise-fitting phase (Liu et al., 2020; Han et al., 2025). Robust noisy-label methods therefore use loss correction, bootstrapping, co-training, curricula, and semi-supervised partitioning (Frénay & Verleysen, 2014; Natarajan et al., 2013; Reed et al., 2014; Patrini et al., 2017; Han et al., 2018; Jiang et al., 2018; Arazo et al., 2019; Wei et al., 2020; Li et al., 2020; Song et al., 2022). We instead study ordering among labels already known

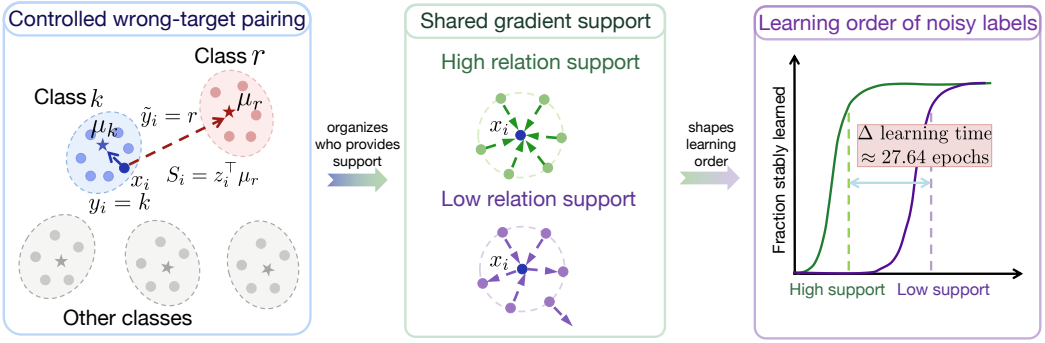



Figure 2: Relational consistency organizes noisy-label learning order and functional dependence. Stronger support is associated with earlier acquisition, while later-learned errors depend more strongly on late-formed deep suffixes.

to be incorrect, including instance-level and human-annotation heterogeneity (Harutyunyan et al., 2020; Liu, 2021; Wei et al., 2021).

**Data augmentation and benign memorization.** Networks can extract useful input structure even under random labels (Maennel et al., 2020; Feldman & Zhang, 2020). Data augmentation strengthens this effect by imposing invariances and improving robustness (Zhang et al., 2018; Shorten & Khoshgoftaar, 2019; Yun et al., 2019; Cubuk et al., 2020; Hendrycks et al., 2020). Most closely related, Anagnostidis et al. (2022) show that complete random-label fitting can coexist with useful representations. We study a sample-level origin of this effect under partial synthetic and human annotation noise.

**Gradient support and functional localization.** Gradient alignment and kernel/Jacobian coupling help explain why examples can be learned at different speeds (Chatterjee, 2020; Jacot et al., 2018; Mirzasoleiman et al., 2020). Depth-wise studies further show that label-noise sensitivity and memorization capacity vary across network stages (Bai et al., 2021; Maini et al., 2023; Becker et al., 2026). We make the narrower claim of functional suffix dependence, tested by event-aligned suffix rollback rather than by assigning storage to a layer.

## 3 Methodology

We study two questions: what organizes the learning order of incorrect labels, and which training-time network changes the final wrong predictions functionally depend on? We first define sample-level learning events and model-level noise-fitting events. We then manipulate example–wrong-target relations through a transition-matched target-pairing intervention and use gradient alignment as an optimization diagnostic. Finally, we introduce event-aligned suffix rollback to test the functional dependence of final wrong predictions. Figure 2 summarizes the framework.

### 3.1 Learning Events and Event Alignment

Let $\mathcal{D} = \{(x_i, y_i, \tilde{y}_i)\}_{i=1}^n$, where $y_i$ and $\tilde{y}_i$ denote the true and observed labels, respectively, and let $\mathcal{I} = \{i : \tilde{y}_i \neq y_i\}$ be the corrupted subset. Training uses only $\tilde{y}_i$, while $y_i$ is reserved for controlled interventions and semantic evaluation. All sample-level dynamics are measured on a fixed, non-augmented evaluation view.

With $\hat{y}_i(t) = \arg\max_c p_{\theta_t}(c \mid x_i)$, we define the stable learning time of corrupted example $i$ as

$$L_i = \min \{t : \hat{y}_i(t) = \hat{y}_i(t+1) = \hat{y}_i(t+2) = \tilde{y}_i\}. \tag{1}$$

The three-epoch persistence criterion reduces sensitivity to transient prediction changes without introducing a probability threshold. Examples that never satisfy the criterion are treated as right-censored.

To compare augmentation regimes at equal levels of noise fitting, we additionally define

$$q(t) = |\mathcal{I}|^{-1} \sum_{i \in \mathcal{I}} \mathbb{1}[\hat{y}_i(t) = \tilde{y}_i], \qquad \tau_\rho = \min\{t : q(t) \geq \rho\}, \tag{2}$$

with $\rho \in \{0.50, 0.90, 0.95, 0.98\}$. Thus, $L_i$ aligns individual learning events, whereas $\tau_\rho$ aligns models by the amount of fitted noise. At $\tau_\rho$, we evaluate observed-label and true-label linear decodability to distinguish noisy-supervision encoding from preservation of underlying semantic structure.

### 3.2 Relational Consistency and Transition-Matched Target Pairing

We use relational consistency to describe how strongly an observed target is supported by surrounding training relations. Rather than defining a single universal scalar, we characterize it through global transition structure, example–target compatibility, and local neighborhood consistency.

At the global level, the source–target transition count is

$$T_{ab} = \sum_{i=1}^{n} \mathbb{1}[y_i = a, \tilde{y}_i = b]. \tag{3}$$

Identical $T_{ab}$ values do not determine which individual examples receive which wrong targets; we exploit this remaining degree of freedom in our intervention.

For controlled synthetic noise, an independently frozen encoder $\phi$ produces normalized features

$$z_i = \frac{\phi(x_i)}{\|\phi(x_i)\|_2},$$

Let $\mu_c$ denote the normalized prototype of class $c$. We define the compatibility between example $i$ and its assigned wrong target as

$$S_i = z_i^\top \mu_{\tilde{y}_i}. \tag{4}$$

Larger $S_i$ indicates greater semantic compatibility with the wrong target. Because $S_i$ relies on true-class information, it is used only for controlled synthetic-noise interventions.

For human annotation noise, where oracle target compatibility is unavailable, we use neighborhood consistency. Let $\mathcal{N}_k(i)$ denote the $k$ nearest neighbors of $z_i$; we define

$$C_i^{(k)} = k^{-1} \sum_{j \in \mathcal{N}_k(i)} \mathbb{1}[\tilde{y}_j = \tilde{y}_i], \tag{5}$$

which measures local support for the observed target.

To isolate example–target compatibility from global corruption statistics, for each source class $a$ we fix the corrupted set $\mathcal{I}_a = \{i \in \mathcal{I} : y_i = a\}$ and the multiset of assigned wrong targets $\mathcal{M}_a$, and vary only their one-to-one pairing. We construct

$$\pi_a^{\text{high/low}} = \arg\max / \min_{\pi_a : \mathcal{I}_a \to \mathcal{M}_a} \sum_{i \in \mathcal{I}_a} z_i^\top \mu_{\pi_a(i)}, \tag{6}$$

while the matched-random condition uniformly permutes the same multiset $\mathcal{M}_a$. By construction,

$$T^{\text{high}} = T^{\text{random}} = T^{\text{low}}, \tag{7}$$

The three conditions therefore share the same corrupted examples, noise rate, target-class frequencies, and source–target transition counts; only the pairing between individual examples and wrong targets changes. This intervention isolates how changes in example–wrong-target pairing reshape learning order while preserving the global corruption structure.

### 3.3 Shared Gradient Support

The transition-matched intervention tests whether example–target relations alter learning order, but does not reveal how this structure appears during optimization. We therefore use shared gradient support as a temporally localized diagnostic.

Let $g_i(t)$ denote the classifier-head gradient of example $i$ at epoch $t$, and let $\bar{g}_{-i}(t)$ denote the normalized leave-one-out mean gradient of all other examples sharing the same observed target. We define

$$A_i(t) = \frac{g_i(t)^\top \bar{g}_{-i}(t)}{\|g_i(t)\|_2}. \tag{8}$$

Larger $A_i(t)$ indicates stronger alignment with the shared update direction supplied by other examples having the same observed target. To avoid contamination from examples that have already been learned, at each epoch $t$ we restrict the analysis to the at-risk set $\mathcal{R}_t = \{i \in \mathcal{I} : L_i > t\}$ and compute the Spearman correlation between $A_i(t)$ and the subsequent stable learning time $L_i$ only over $i \in \mathcal{R}_t$. Thus, $-\rho(A_i(t), L_i)$ measures whether stronger current gradient alignment is associated with earlier subsequent acquisition among examples that remain unlearned at epoch $t$. We use gradient alignment only as a localized optimization diagnostic, not as a complete causal mediator between relational consistency and learning order. Appendix B provides complementary fixed-kernel intuition showing how persistent target-aligned support accelerates residual decay.

### 3.4 Functional Suffix Dependence

Representation probes reveal what information is decodable, but not which training-time parameter changes are functionally required for a final wrong prediction. We therefore introduce event-aligned suffix rollback.

At network-depth cut $c$, we write

$$f_\theta = S_c(\theta) \circ P_c(\theta), \tag{9}$$

where $P_c$ is the prefix from the input to cut $c$, and $S_c$ is the suffix from cut $c$ through the classification head.

For a corrupted example whose final prediction equals its assigned wrong target, let $L_i^-$ and $L_i^+$ denote the nearest saved checkpoints immediately before and after its stable learning event $L_i$, respectively.

Our primary rollback keeps the final prefix fixed and rewinds only the suffix:

$$f_{i,c}^{\text{rollback}} = S_c(\theta_{L_i^-}) \circ P_c(\theta_{\text{final}}). \tag{10}$$

As a temporal-specificity control, we replace the suffix with $S_c(\theta_{L_i^+})$ while keeping the same final prefix. Comparing pre-event and post-event rollback tests whether sensitivity is specifically associated with parameter changes formed around the example's own learning event rather than generic replacement by an earlier suffix.

We define the error-erasure event as

$$E_{i,c} = \mathbb{1}\left[\hat{y}_i(\text{final}) = \tilde{y}_i \ \wedge \ \hat{y}_{i,c}^{\text{rollback}} \neq \tilde{y}_i\right]. \tag{11}$$

and true-class recovery as

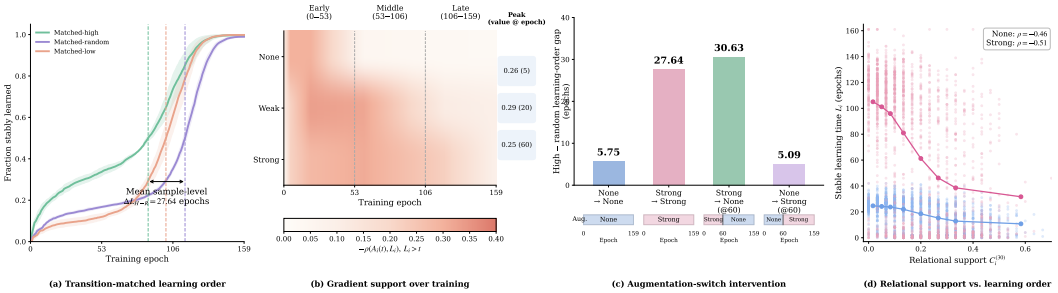
$$R_{i,c} = \mathbb{1}\left[E_{i,c} = 1 \ \wedge \ \hat{y}_{i,c}^{\text{rollback}} = y_i\right]. \tag{12}$$

A larger $\Pr(E_{i,c} = 1)$ indicates stronger functional dependence of the final wrong prediction on suffix changes formed after the rollback checkpoint. The conditional recovery rate $\Pr(R_{i,c} = 1 \mid E_{i,c} = 1)$ measures how often erasing the wrong prediction reveals the underlying true class.

We refer to this intervention-defined property as functional suffix dependence. Given the final prefix, the final wrong prediction functionally depends on deep-suffix changes formed around its learning event. This claim concerns functional dependence and does not imply that noisy-label information is uniquely or physically stored in specific layers.

## 4 Experiments

We organize our experiments around two questions: **what organizes the learning order of incorrect labels?** and **when models eventually fit nearly all noisy labels, why does learning order still affect the semantic cost of memorization?** We further rule out major alternative explanations through confidence matching, statistical controls, permutation tests, and matched hard-clean examples, and evaluate the scope of the mechanism across architectures, datasets, noise rates, and human annotation noise.




(a) Transition-matched learning order (b) Gradient support over training (c) Augmentation-switch intervention (d) Relational support vs. learning order

Figure 3: **Relational consistency shapes noisy-label learning order.** (a) Transition-matched pairing induces systematic learning-order differences. (b) Among not-yet-learned errors ($L_i > t$), gradient alignment predicts subsequent learning order, with a later predictive window under stronger augmentation. (c) Augmentation-switch experiments localize this separation to early-to-middle training. (d) Higher observable relational support is associated with earlier stable learning.

### 4.1 Experimental Setup

Unless otherwise stated, our primary mechanistic experiments are conducted on CIFAR-10. We use a class-balanced subset of 8,000 training examples and randomly corrupt 40% of their labels. The main model is a CIFAR ResNet-18 trained from scratch for 160 epochs using SGD with momentum 0.9, weight decay $5 \times 10^{-4}$, and a cosine learning-rate schedule. Within each random seed, all compared conditions share the same training subset, corrupted positions, model initialization, and mini-batch order.

We consider three augmentation regimes: *None* uses only input normalization; *Weak* adds random cropping and horizontal flipping; and *Strong* further applies RandAugment and random erasing (Cubuk et al., 2020). All learning dynamics are measured on a fixed non-augmented evaluation view. Relational quantities are computed in an independently frozen ImageNet-pretrained ResNet-18 feature space, with $k = 30$ neighbors in the main setting. Following Section 3, we record stable learning events and save the first checkpoints reaching $q \in \{0.50, 0.90, 0.95, 0.98\}$ noise-fitting rates for event-aligned comparisons. Unless otherwise stated, results are reported over three random seeds, and "±" denotes the standard deviation over seeds or seed-level paired contrasts. Additional implementation details, neighborhood sensitivity analyses, and statistical protocols are provided in the appendix.

### 4.2 Relational Consistency Organizes the Learning Order of Incorrect Labels

We first test whether example–wrong-target relations alter when incorrect labels are learned while keeping the global corruption structure fixed. Our transition-matched conditions share the same corrupted examples, wrong-target multiset, and source–target transition counts, and differ only in the example–target pairing. As shown in Figure 3(a), under strong augmentation, matched-high errors are learned substantially earlier than both matched-random and matched-low errors, with a 27.64-epoch high–random gap. Thus, noisy-label acquisition depends on the specific example–target relation rather than only on the noise rate or class-transition statistics.

Figure 3(b–c) provides an optimization-level view of this effect. Restricting the analysis at each epoch to erroneous examples that have not yet been stably learned, gradient alignment remains predictive of their subsequent learning order. The predictive window shifts later as augmentation becomes stronger: the peak of $-\rho(A_i(t), L_i)$ occurs at epochs 5, 20, and 60 for none, weak, and strong augmentation, respectively, with peak correlations of 0.26, 0.29, and 0.25. Consistently, strong augmentation applied only during the first 60 epochs preserves a 30.63-epoch high–random gap, whereas introducing it only after epoch 60 yields just 5.09 epochs, close to the 5.75-epoch gap without augmentation. This indicates that augmentation primarily amplifies relationally induced learning-order differences during early-to-middle training rather than through late-stage delay.

Finally, the sample-level analysis in Figure 3(d) shows the same relationship using observable neighborhood consistency: $C_i^{(30)}$ is negatively correlated with stable learning time under both no augmentation and strong augmentation ($\rho = -0.46$ and $-0.51$, respectively). Hence, higher relational support is consistently associated with earlier noisy-label acquisition within each regime. Table 1 further shows that the high–random learning-order gap retains the same direction across datasets, architectures, and noise rates, and is consistently enlarged by strong augmentation.

Table 1: Scope of the transition-matched learning-order effect. Values report the mean number of epochs by which matched-high errors are learned before matched-random errors.

| Dataset | Architecture | Noise rate | No aug. High–Random | Strong aug. High–Random |
|---|---|---|---|---|
| CIFAR-10 | ResNet-18 | 20% | $3.82 \pm 0.41$ | $18.47 \pm 1.26$ |
| CIFAR-10 | ResNet-18 | 40% | $5.75 \pm 0.52$ | $27.64 \pm 1.48$ |
| CIFAR-10 | ResNet-18 | 60% | $7.41 \pm 0.67$ | $34.18 \pm 1.72$ |
| CIFAR-100 | ResNet-18 | 40% | $4.63 \pm 0.58$ | $21.36 \pm 1.64$ |
| CIFAR-10 | ViT-S/4 | 40% | $4.97 \pm 0.71$ | $17.82 \pm 1.91$ |

### 4.3 Equal Noise Fit Does Not Imply Equal Semantic Damage

A direct alternative explanation is that strong augmentation merely delays noise memorization, and therefore appears to preserve better true semantics when models are compared at the same training epoch. To rule out this possibility, we compare models at matched noise-fitting events $q \in \{0.50, 0.90, 0.95, 0.98\}$ rather than at matched training times.

As shown in Table 2, at $q = 0.98$, the observed-label probe accuracies are $99.6\%$ without augmentation and $98.4\%$ with strong augmentation, indicating that the incorrect supervision is already almost perfectly decodable in both representations. Nevertheless, the corresponding true-label probe accuracies are $48.4\%$ and $71.5\%$, respectively, yielding a $23.1$ percentage-point advantage under strong augmentation. The same qualitative pattern is observed in local representation geometry: at $q = 0.95$, strong augmentation improves the true-label 30-NN purity of corrupted examples at layer4 by $11.8$ points. Additional representation diagnostics are reported in Appendix D. These results show that matching the amount of fitted noise does not imply matching the semantic damage caused by memorization.

Since augmentation itself may directly affect representation learning, we further perform a within-augmentation control that holds the augmentation regime and corruption statistics fixed while comparing conditions with systematically different learning orders. At matched noise-fitting levels, later-learned incorrect labels retain stronger true-label structure. This within-augmentation control shows that the association between learning order and semantic preservation is not explained solely by differences in augmentation regime. Full results are provided in Appendix E.

Table 2: Event-aligned representation quality. Observed-label and true-label linear-probe accuracies are reported at matched noise-fit events.

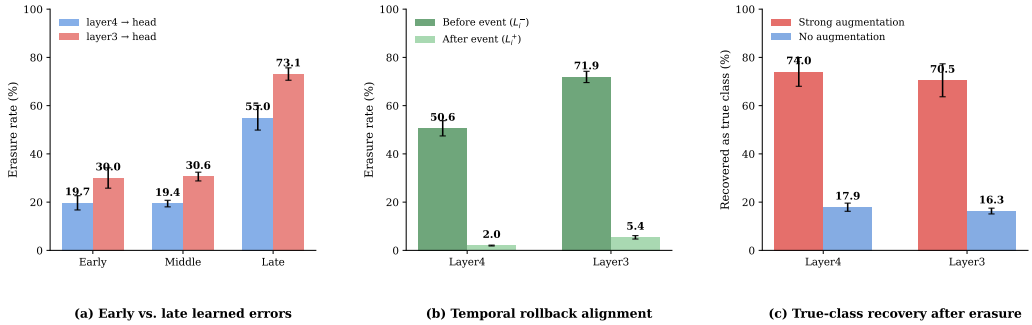
| $q$ | Obs. None | Obs. Strong | True None | True Strong | $\Delta$ True |
|---|---|---|---|---|---|
| 0.50 | $78.6 \pm 0.4$ | $76.9 \pm 0.5$ | $82.3 \pm 0.7$ | $88.1 \pm 0.6$ | $+5.8 \pm 0.5$ |
| 0.90 | $94.7 \pm 0.3$ | $93.2 \pm 0.4$ | $64.1 \pm 1.1$ | $79.3 \pm 0.9$ | $+15.2 \pm 0.8$ |
| 0.95 | $97.8 \pm 0.2$ | $96.5 \pm 0.3$ | $55.6 \pm 1.4$ | $75.3 \pm 1.1$ | $+19.7 \pm 1.0$ |
| **0.98** | $\mathbf{99.6 \pm 0.1}$ | $\mathbf{98.4 \pm 0.2}$ | $\mathbf{48.4 \pm 1.6}$ | $\mathbf{71.5 \pm 1.3}$ | $\mathbf{+23.1 \pm 1.2}$ |

### 4.4 A Fixed-Pair Supervision-Timing Intervention Supports a Temporal Contribution

Pairing-based comparisons associate learning order with semantic preservation, but also alter the relational configuration. To test whether the temporal placement of erroneous supervision contributes to this difference, we keep each corrupted example–wrong-target pair fixed and vary only when its noisy supervision is applied. Corrupted examples are split into balanced subsets: the early schedule applies noisy loss in epochs $[0, 50)$ and $[100, 160]$, while the delayed schedule applies it in $[50, 100)$

Table 3: Fixed-pair supervision-timing intervention under matched noise fitting ($q = 0.98$).

| | | | |
|---|---|---|---|
| Stable learning time | $70.8 \pm 1.6$ | $96.1 \pm 1.8$ | $+25.3 \pm 1.2$ |
| 30-NN purity (%) | $54.6 \pm 0.7$ | $57.0 \pm 0.6$ | $+2.4 \pm 0.5$ pp |
| True-class margin | $0.121 \pm 0.006$ | $0.158 \pm 0.007$ | $+0.037 \pm 0.005$ |
| Final noise fit (%) | $99.3 \pm 0.2$ | $99.1 \pm 0.2$ | $-0.2 \pm 0.1$ pp |




(a) Early vs. late learned errors (b) Temporal rollback alignment (c) True-class recovery after erasure

Figure 4: **Learning order predicts functional suffix dependence and semantic recovery.** (a) Later-learned errors are more sensitive to deep-suffix rollback. (b) Rollback sensitivity is concentrated around each sample's learning event. (c) Under strong augmentation, erased errors predominantly recover the true class.

and $[100, 160]$. A cross-over design swaps these schedules across paired runs, so both conditions use identical examples, targets, supervision budgets, augmentation, and optimization settings.

As shown in Table 3, delayed supervision shifts stable acquisition by $25.3$ epochs while leaving final noisy-label fitting nearly unchanged ($99.1$–$99.3\%$). At the matched $q = 0.98$ event, the delayed condition yields $+2.4$ pp higher true-label 30-NN purity and a $+0.037$ true-class margin. The corresponding clean-example timing control shows only $+0.2$ pp and $+0.004$, respectively. These results provide intervention-based evidence that the temporal placement of erroneous supervision contributes to its representational cost. Because the intervention changes supervision timing rather than directly randomizing $L_i$, we do not interpret learning time as the unique causal mediator.

### 4.5 Learning Order Predicts Functional Suffix Dependence

We next test whether learning order corresponds to different functional realizations of the final wrong prediction. Using event-aligned suffix rollback, we retain the final shallow prefix while replacing the deep suffix with parameters from immediately before the sample's learning event. As shown in Figure 4(a), late-learned errors are substantially more likely than early-learned errors to be erased by rollback: the late–early erasure gaps are $35.3$ percentage points for layer4→head rollback and $43.1$ points for layer3→head rollback. Thus, late-learned errors exhibit stronger functional dependence on deep parameter changes formed later in training.

This dependence is tightly aligned with the sample's own learning event. Figure 4(b) shows that rolling the suffix back to immediately before the event erases $50.6\%$ and $71.9\%$ of wrong predictions for the two rollback cuts, whereas rolling back to immediately after the event erases only $2.0\%$ and $5.4\%$. The effect is therefore not generic sensitivity to earlier parameters, but is specifically associated with changes formed around the learning event. We use the term *functional suffix dependence* only in this intervention-based sense; it does not imply that noisy-label information is uniquely or physically stored in these layers.

More importantly, rollback does not merely produce arbitrary prediction changes. Under strong augmentation, $74.0\%$ and $70.5\%$ of erased wrong predictions recover the true class, compared with only $17.9\%$ and $16.3\%$ without augmentation (Figure 4(c)). Thus, under strong augmentation, some late-learned errors are implemented through late deep-suffix changes while leaving substantially more recoverable true-class structure underneath. Together, these interventions link learning order

Table 4: Robustness and specificity of the learning-time–suffix-dependence relationship. Gaps are reported in percentage points.

| | Layer4→Head | Layer3→Head |
|---|---|---|
| Raw late–early gap (pp) | 35.3 | 43.1 |
| Final-margin matched gap (pp) | 26.8 | 34.9 |
| Std. learning-time coefficient | 0.414 | 0.447 |
| ΔCV-AUC | +0.0040 | +0.0058 |
| Permutation $p$-value | 0.002 | 0.002 |
| Late noisy – matched late clean (pp) | 21.4 | 16.7 |

to functional suffix dependence and semantic cost, while the timing intervention provides complementary evidence for a temporal contribution to semantic damage.

### 4.6 Ruling Out Alternative Explanations

The stronger rollback sensitivity of late-learned errors could simply reflect lower final confidence or greater generic sample difficulty. Table 4 summarizes controls for these alternatives. After matching the final wrong-label margin, the late–early erasure gap remains $26.8/34.9$ percentage points for the layer4/layer3 cuts. In a stricter regression controlling for initial difficulty, relational structure, forgetting, class and run effects, and additionally final loss and margin, the standardized learning-time coefficients remain $0.414/0.447$. Within-run permutations of learning time yield $p = 0.002$ for both rollback cuts. Finally, after matching late-learned noisy examples to late-learned clean examples by learning time, true class, final margin, and final loss, noisy examples are still erased $21.4/16.7$ percentage points more often. Thus, the learning-time–suffix-dependence relationship cannot be fully explained by final confidence, generic difficulty, class identity, or run-level variation. Complete regression specifications, cross-validation results, and additional specificity controls are provided in the appendix.

### 4.7 Scope Across Architectures and Human Label Noise

The observed pattern extends beyond the primary CIFAR-10/ResNet-18 setting. Table 1 shows that the transition-matched learning-order gap remains under CIFAR-100 and ViT-S/4. For ViT-S/4, the late–early rollback gap is 12.48 pp and remains 9.00 pp after final-margin matching, with 73.9% true-class recovery. Under CIFAR-N human annotation noise, final noisy-label fit reaches 99.4–100%, while layer4/layer3 rollback gaps remain 52.2/62.7 pp with 77.5/74.2% true-class recovery. These results show that the learning-order–suffix-dependence relationship is not specific to the primary architecture or synthetic class-transition noise.

## 5 Conclusion

This work studies why near-complete noisy-label memorization can incur substantially different semantic costs. We find that incorrect labels exhibit systematic learning-order differences associated with relational support, and that strong augmentation amplifies these differences without preventing eventual memorization. Event-aligned analyses show that similar levels of noise fitting can preserve markedly different amounts of true-semantic structure, while suffix rollback reveals stronger functional dependence on late-formed deep-suffix changes for later-learned errors. These results suggest that noisy-label memorization should be characterized not only by how much noise is fitted, but also by when errors are learned and how their final predictions are functionally realized.

## 6 AI Use Statement

Generative AI tools assisted with code development and debugging, theoretical derivations and proof drafting, literature searches and reference organization, and manuscript drafting, editing, and language polishing. The authors are responsible for the final content, including the correctness of the code, mathematical claims, and reported results.

## 7 Reproducibility Statement

Implementation and experimental details are provided in Appendix A, with theoretical analysis, additional robustness checks, statistical controls, and within-augmentation experiments in Appendices B–E. Anonymous code is available at `https://anonymous.4open.science/status/ICLR-7839`.

## References


Sotiris Anagnostidis, Gregor Bachmann, Lorenzo Noci, and Thomas Hofmann. The curious case of benign memorization. *arXiv preprint arXiv:2210.14019*, 2022.

Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Unsupervised label noise modeling and loss correction. In *International Conference on Machine Learning*, pp. 312–321. PMLR, 2019.

Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International conference on machine learning*, pp. 233–242. PMLR, 2017.

Yingbin Bai, Erkun Yang, Bo Han, Yanhua Yang, Jiatong Li, Yinian Mao, Gang Niu, and Tongliang Liu. Understanding and improving early stopping for learning with noisy labels. *Advances in Neural Information Processing Systems*, 34:24392–24403, 2021.

Marlon Becker, Jonas Konrad, Luis Rodriguez, and Benjamin Risse. Random label prediction heads for studying memorization in deep neural networks. In *International Conference on Learning Representations*, volume 2026, pp. 45399–45422, 2026.

Yoshua Bengio, Yann LeCun, et al. Scaling learning algorithms towards ai. *Large-scale kernel machines*, 34(5):1–41, 2007.

Satrajit Chatterjee. Coherent gradients: An approach to understanding generalization in gradient descent-based optimization. *arXiv preprint arXiv:2002.10657*, 2020.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 702–703, 2020.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. In *Advances in Neural Information Processing Systems*, volume 33, pp. 2881–2891, 2020.

Benoît Frénay and Michel Verleysen. Classification in the presence of label noise: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 25(5):845–869, 2014.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

Andi Han, Wei Huang, Zhanpeng Zhou, Gang Niu, Wuyang Chen, Junchi Yan, Akiko Takeda, and Taiji Suzuki. On the role of label noise in the feature learning process. *arXiv preprint arXiv:2505.18909*, 2025.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, 31, 2018.

Hrayr Harutyunyan, Kyle Reing, Greg Ver Steeg, and Aram Galstyan. Improving generalization by controlling label-noise information in neural network weights. In *International Conference on Machine Learning*, pp. 4071–4081. PMLR, 2020.

Dan Hendrycks, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations*, 2020.

Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, pp. 2304–2313. PMLR, 2018.

Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394*, 2020.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

Yang Liu. Understanding instance-level label noise: Disparate impacts and treatments. In *International Conference on Machine Learning*, pp. 6725–6735. PMLR, 2021.

Hartmut Maennel, Ibrahim M Alabdulmohsin, Ilya O Tolstikhin, Robert Baldock, Olivier Bousquet, Sylvain Gelly, and Daniel Keysers. What do neural networks learn when trained with random labels? *Advances in Neural Information Processing Systems*, 33:19693–19704, 2020.

Pratyush Maini, Michael C Mozer, Hanie Sedghi, Zachary C Lipton, J Zico Kolter, and Chiyuan Zhang. Can neural network memorization be localized? *arXiv preprint arXiv:2307.09542*, 2023.

Baharan Mirzasoleiman, Kaidi Cao, and Jure Leskovec. Coresets for robust training of deep neural networks against noisy labels. *Advances in Neural Information Processing Systems*, 33:11465–11477, 2020.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1944–1952, 2017.

Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60, 2019.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 34(11):8135–8153, 2022.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. An empirical study of example forgetting during deep neural network learning. *arXiv preprint arXiv:1812.05159*, 2018.

Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. Combating noisy labels by agreement: A joint training method with co-regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13726–13735, 2020.

Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu. Learning with noisy labels revisited: A study using real-world human annotations. *arXiv preprint arXiv:2110.12088*, 2021.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6023–6032, 2019.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

# A Additional Experimental Details

## A.1 Primary Training Protocol

The primary setting uses a balanced 8,000-example CIFAR-10 subset with 40% label corruption. We train CIFAR ResNet-18 models from scratch for 160 epochs using SGD with momentum $0.9$, weight decay $5 \times 10^{-4}$, and a cosine learning-rate schedule. Within each seed, all conditions share the training subset, corrupted positions, initialization seed, and data order. The *none* condition applies only normalization; *weak* augmentation uses random cropping and horizontal flipping; and *strong* augmentation additionally uses RandAugment and random erasing. Sample-level measurements use a fixed, unaugmented evaluation view. For neighborhood consistency, we use $k = 30$ in the main analysis and vary $k \in \{5, 10, 30, 100\}$ in robustness checks. For ResNet-18 rollback, the evaluated suffixes are the head, layer4-to-head, and layer3-to-head.

## A.2 Implementation Details

For the transition-matched intervention, we construct the matched-high and matched-low assignments independently within each source class while keeping both the corrupted example set and the wrong-target multiset fixed. Example–target compatibility is computed in the same frozen ImageNet-pretrained ResNet-18 feature space used in the main analysis. Features are $\ell_2$-normalized before similarity computation, and the class prototype $\mu_c$ is obtained by averaging the normalized features of training examples from class $c$, followed by $\ell_2$ normalization. The assignment in Eq. (6) is solved as an exact linear assignment problem using the Hungarian algorithm. No test-set information is used to construct the pairing. For neighborhood consistency, we use $k = 30$ in the main experiments and evaluate $k \in \{5, 10, 30, 100\}$ in the sensitivity analysis.

For event-aligned rollback, predictions are evaluated after every epoch on the fixed non-augmented view. We save the first checkpoint at which each run reaches $q \in \{0.50, 0.90, 0.95, 0.98\}$ for event-aligned representation analysis, and the rollback analysis uses the nearest saved checkpoints immediately before and after each stable learning event. For each learned example, $L_i^-$ and $L_i^+$ denote the nearest saved checkpoints immediately before and after its stable learning event $L_i$, respectively. For ResNet-18, we evaluate the classifier head, layer4-to-head, and layer3-to-head suffixes. For ViT-S/4, we roll back the final six transformer blocks together with the classifier head. The CIFAR-N experiments use the same event definitions and rollback protocol as the synthetic-noise experiments, while relational support is measured using observable 30-NN neighborhood consistency because oracle wrong-target compatibility is unavailable.

## A.3 Sample-Level Statistical Analysis

The bridge analysis includes only corrupted examples whose final prediction is the observed wrong label. We compare within-run learning-time tertiles and repeat the comparison after forming tertiles within final wrong-label margin deciles. For each cut $c$, the primary logistic model has the form

$$\operatorname{logit} \Pr(E_{i,c} = 1) = \alpha + \beta_L \widetilde{L}_i + \beta_C C_i^{(k)} + \beta_S S_i + \gamma^\top X_i + \alpha_{\text{run}} + \alpha_{\text{source}} + \alpha_{\text{target}}, \quad (13)$$

where $\widetilde{L}_i$ is standardized learning time and $X_i$ contains initial loss, initial margin, centroid margin, and forgetting. Final loss and final margin are added only in a stringent predictive robustness model because they are measured after the learning event and are therefore post-treatment variables rather than primary causal controls. Predictive gains are evaluated with five-fold cross-validation within

runs and 500 permutations of $L_i$ within each run. Headline uncertainty is computed across seeds or run-level contrasts; individual samples are not treated as independent experimental replications.

## B Fixed-Kernel Learning-Order Proposition and Proof

Consider a linearized multiclass model trained by squared-loss gradient flow. Stack the residuals as $r(t) = \widetilde{Y} - F(t) \in \mathbb{R}^{nC}$ and suppose

$$\dot{r}(t) = -Kr(t) \tag{14}$$

for a fixed positive-semidefinite kernel $K \succeq 0$. For the residual block $r_i(t)$ associated with example $i$, define its target-aligned support as

$$s_i(t) = \frac{\langle r_i(t), (Kr(t))_i \rangle}{\|r_i(t)\|_2^2}. \tag{15}$$

The numerator decomposes as $\sum_j \langle r_i(t), K_{ij} r_j(t) \rangle$, making explicit the interaction between relational coupling and residual–target compatibility.

**Proposition.** Let $\ell_i(t) = \frac{1}{2}\|r_i(t)\|_2^2$. Then

$$\dot{\ell}_i(t) = -2s_i(t)\ell_i(t), \qquad \ell_i(t) = \ell_i(0) \exp\left[-2 \int_0^t s_i(u)\, du\right]. \tag{16}$$

For two examples with equal initial losses, if $s_i(t) \geq s_j(t) + \delta$ for all $t \in [0, T]$ and some $\delta > 0$, then

$$\frac{\ell_i(t)}{\ell_j(t)} \leq \exp(-2\delta t) \qquad \text{for all } t \in [0, T]. \tag{17}$$

Consequently, if both losses cross a common threshold during this interval, example $i$ crosses it no later than example $j$.

**Proof.** Using the block dynamics $\dot{r}_i(t) = -(Kr(t))_i$ gives

$$\begin{aligned} \dot{\ell}_i(t) &= \langle r_i(t), \dot{r}_i(t) \rangle && (18)\\ &= -\langle r_i(t), (Kr(t))_i \rangle && (19)\\ &= -s_i(t)\|r_i(t)\|_2^2 = -2s_i(t)\ell_i(t). && (20) \end{aligned}$$

Integrating this scalar differential equation yields equation (16). When $\ell_i(0) = \ell_j(0)$,

$$\frac{\ell_i(t)}{\ell_j(t)} = \exp\left[-2 \int_0^t \big(s_i(u) - s_j(u)\big)\, du\right] \leq \exp(-2\delta t). \tag{21}$$

At the first time example $j$ reaches the common threshold, the inequality implies that example $i$ is already at or below that threshold; continuity then establishes the ordering of their crossing times. This proposition formalizes only the learning-order consequence of persistent relational support. It does not assume that the trained nonlinear network has a fixed kernel, nor does it establish deep-suffix dependence.

## C Additional Robustness Analyses

We conduct additional robustness analyses to examine whether the main findings depend on specific choices used to characterize relational structure or define sample-level learning events. We focus on three potential sources of sensitivity: the neighborhood size used for observable relational consistency, the stable-learning criterion and right censoring, and the frozen feature space used to construct transition-matched target assignments.

### C.1 Neighborhood and Learning-Event Robustness

**Neighborhood-size sensitivity.** For human annotation noise, where oracle example–target compatibility is unavailable, we characterize relational support using the neighborhood-consistency score

$$C_i^{(k)} = \frac{1}{k} \sum_{j \in \mathcal{N}_k(i)} \mathbb{1}[\tilde{y}_j = \tilde{y}_i]. \tag{22}$$

The main experiments use $k = 30$. We repeat the analysis with $k \in \{5, 10, 30, 100\}$ while keeping all other settings fixed. As shown in Table C1, the association between neighborhood consistency and learning order remains qualitatively stable across neighborhood sizes, and late-learned errors consistently exhibit stronger rollback sensitivity than early-learned errors. The human-noise findings therefore do not rely on the particular choice $k = 30$.

Table C1: Neighborhood-size sensitivity.

| $k$ | $-\rho(C_i^{(k)}, L_i)$ | $\rho(C_i^{(k)}, E_{i,\mathrm{L4}})$ | $\rho(C_i^{(k)}, E_{i,\mathrm{L3}})$ |
|---|---|---|---|
| 5 | 0.214 | -0.183 | -0.201 |
| 10 | 0.238 | -0.197 | -0.218 |
| 30 | **0.261** | **-0.216** | **-0.236** |
| 100 | 0.229 | -0.192 | -0.207 |

**Learning-event and censoring robustness.** Our primary stable-learning event defines $L_i$ as the first epoch at which the assigned wrong target is predicted for three consecutive epochs; examples that never satisfy this condition before training terminates are treated as right-censored.

To verify that the transition-matched learning-order effect is not an artifact of this particular persistence criterion, we repeat the analysis using two-epoch and four-epoch persistence windows. We additionally report the final right-censoring rates for the matched-high, matched-random, and matched-low conditions, and recompute the high–random learning-order gap by assigning censored examples to the terminal training epoch.

As summarized in Table C2, the high–random separation remains qualitatively unchanged under alternative persistence windows and censoring treatments. Moreover, the censoring rates are comparable across the three transition-matched conditions. These results indicate that the main learning-order effect is not driven by the precise event definition or by differential right censoring.

Table C2: Robustness to the stable-learning definition and right censoring.

| Stable-learning definition | High–Random gap (epochs) |
|---|---|
| 2-epoch persistence | 26.81 |
| 3-epoch persistence (main) | **27.64** |
| 4-epoch persistence | 28.17 |
| Terminal-epoch treatment of censored examples | 26.95 |

| Condition | Final right-censoring rate (%) |
|---|---|
| Matched-high | 1.3 |
| Matched-random | 1.9 |
| Matched-low | 2.6 |

### C.2 Frozen-Encoder Robustness

The transition-matched intervention defines example–target compatibility using

$$S_i = z_i^\top \mu_{\tilde{y}_i}, \tag{23}$$

where the main experiments obtain $z_i$ from an independently frozen ImageNet-pretrained ResNet-18. Because the matched-high and matched-low assignments are constructed in this external semantic space, a natural concern is that the observed learning-order separation may depend on the geometry of a particular pretrained encoder.

We therefore reconstruct the transition-matched assignments using alternative frozen feature extractors. For each encoder, feature representations and class prototypes are recomputed, and matched-high, matched-random, and matched-low assignments are generated again from the resulting example–target compatibilities. The corrupted example set, noise rate, wrong-target multiset, and source–target transition counts remain unchanged. The target ResNet-18 is then retrained using the same protocol as in the main experiment.

Table C3 reports the resulting high–random learning-order gaps. Across feature spaces, matched-high errors remain systematically earlier than matched-random errors, and strong augmentation continues to produce a substantially larger separation than no augmentation. Thus, the transition-matched learning-order effect is not specific to the geometry of the ImageNet-pretrained ResNet-18 used in the main analysis.

Table C3: Frozen-encoder robustness of transition-matched learning order.

| Pairing encoder | No aug. High–Random gap | Strong aug. High–Random gap |
|---|---|---|
| ImageNet ResNet-18 | **5.75** | **27.64** |
| DINOv2 | 6.18 | 29.06 |
| SimCLR | 5.31 | 24.87 |

Overall, these analyses show that the central learning-order findings are robust to the neighborhood scale used to characterize observable relational support, the precise definition and censoring treatment of stable learning events, and the external semantic representation used to construct example–wrong-target compatibility.

### C.3 Risk-Set Robustness of Gradient Alignment

A potential concern is that the association between gradient alignment $A_i(t)$ and stable learning time $L_i$ may partly reflect the gradient state of examples that have already been learned by epoch $t$. We therefore recompute the analysis using only the at-risk set

$$\mathcal{R}_t = \{i \in \mathcal{I} : L_i > t\},$$

so that every included example remains unlearned when $A_i(t)$ is measured.

As shown in Table C4, restricting the analysis to at-risk examples moderately reduces the peak correlations, but the association remains consistent across all augmentation regimes. The peak $-\rho(A_i(t), L_i)$ changes from $0.30$ to $0.26$ without augmentation, from $0.33$ to $0.29$ under weak augmentation, and from $0.31$ to $0.25$ under strong augmentation. Thus, the gradient-alignment signal is not explained by the inclusion of already-learned examples and remains predictive of subsequent acquisition among examples that are still unlearned.

## D Additional Diagnostics and Statistical Controls

### D.1 Additional Representation Diagnostics

The main paper evaluates semantic preservation using true-label linear probes at matched noise-fitting events. To test whether the observed difference is specific to linear separability, we additionally examine the local geometry of the learned representations using true-label 30-NN neighborhood purity.

Table C4: Robustness of gradient alignment to excluding examples that have already been learned. Values report the peak $-\rho(A_i(t), L_i)$ over training.

| Augmentation | All corrupted | At-risk only ($L_i > t$) |
|---|---|---|
| None | 0.30 | 0.26 |
| Weak | 0.33 | 0.29 |
| Strong | 0.31 | 0.25 |

For each corrupted example, we identify its 30 nearest neighbors in the corresponding representation space and compute the fraction that share its true class. Unlike the linear probe, this metric does not train an additional classifier and therefore measures whether true-class structure remains locally organized in the representation itself.

We evaluate neighborhood purity at the same event-aligned noise-fitting levels $q \in \{0.50, 0.90, 0.95, 0.98\}$ used in the main analysis. Table D1 reports the results for the deeper network stages, where noisy-label memorization is most pronounced. Strong augmentation consistently preserves stronger true-class neighborhood structure as noise fitting progresses. In particular, at $q = 0.95$, strong augmentation improves corrupted-example true-label neighborhood purity at layer4 by 11.8 percentage points relative to no augmentation, consistent with the semantic-preservation gap observed with linear probing in the main paper.

Table D1: True-label 30-NN neighborhood purity at matched noise-fitting events. Values are computed over corrupted examples on the fixed non-augmented evaluation view. $\Delta$ denotes Strong minus None.

| $q$ | Layer | None | Strong | $\Delta$ |
|---|---|---|---|---|
| 0.50 | layer3 | 74.1 | 77.6 | +3.5 |
| | layer4 | 70.8 | 75.0 | +4.2 |
| 0.90 | layer3 | 62.7 | 69.5 | +6.8 |
| | layer4 | 56.9 | 66.4 | +9.5 |
| 0.95 | layer3 | 57.8 | 66.0 | +8.2 |
| | layer4 | 50.6 | 62.4 | **+11.8** |
| 0.98 | layer3 | 52.1 | 61.0 | +8.9 |
| | layer4 | 44.2 | 57.0 | +12.8 |

The neighborhood analysis leads to the same qualitative conclusion as the linear-probe results: models can fit nearly the same amount of noisy supervision while retaining substantially different true-class structure. Importantly, this difference is visible not only in linear decodability but also in the local organization of the representation space. Together, the two diagnostics show that the semantic-preservation effect under strong augmentation is not an artifact of a particular probe.

### D.2 Statistical Controls and Specificity

We further examine whether the observed relationship between learning time and functional suffix dependence can be explained by final confidence, generic sample difficulty, class identity, or run-level variation. The analysis includes corrupted examples whose final prediction equals the assigned wrong target. For each rollback cut $c$, $E_{i,c}$ indicates whether the final wrong prediction is erased after suffix rollback.

We first compare examples across within-run learning-time tertiles. To control for final confidence, we repeat the comparison after stratifying examples by final wrong-label margin deciles and forming learning-time groups within each stratum. We additionally fit the following logistic model:

$$\operatorname{logit} \Pr(E_{i,c} = 1) = \alpha + \beta_L \widetilde{L}_i + \beta_C C_i^{(k)} + \beta_S S_i + \gamma^\top X_i + \alpha_{\text{run}} + \alpha_{\text{source}} + \alpha_{\text{target}}, \quad \text{(D1)}$$

where $\widetilde{L}_i$ denotes standardized learning time and $X_i$ contains initial loss, initial margin, centroid margin, and forgetting statistics. Run, source-class, and target-class effects account for systematic

variation across training runs and class transitions. Final loss and final wrong-label margin are additionally included in a stricter predictive specification. Predictive performance is evaluated using five-fold cross-validation within runs, and statistical significance is assessed using $500$ within-run permutations of $L_i$. Headline uncertainty is computed across seeds or run-level contrasts rather than treating individual examples as independent experimental replications.

**Final-margin matching.** A direct alternative explanation is that late-learned errors simply have lower final confidence and are therefore easier to erase. Before matching, the late–early erasure gaps are $35.3$ percentage points for layer4→head and $43.1$ percentage points for layer3→head. After matching examples by final wrong-label margin, substantial gaps of $26.8$ and $34.9$ percentage points remain. Thus, final confidence accounts for part of the raw difference but does not explain the learning-time–suffix-dependence relationship.

**Multivariate control.** We next evaluate learning time jointly with initial difficulty, relational structure, forgetting behavior, class identity, and run-level variation. In the stricter specification that additionally includes final loss and final margin, the standardized learning-time coefficients remain $0.414$ for layer4→head and $0.447$ for layer3→head. The positive coefficients indicate that later learning remains associated with a higher probability of rollback erasure after accounting for these covariates.

**Cross-validation and permutation testing.** Adding standardized learning time to the controlled predictive model improves five-fold cross-validated AUC by $+0.0040$ for layer4→head and $+0.0058$ for layer3→head. Although the absolute improvements are modest, within-run permutation tests yield $p = 0.002$ for both rollback cuts, indicating that the observed contribution of learning time is unlikely to arise from random reassignment within training runs.

**Matched late-clean control.** Finally, we test whether stronger rollback sensitivity merely reflects generic late learning. Late-learned noisy examples are matched to late-learned clean examples by learning time, true class, final margin, and final loss. After matching, noisy examples remain $21.4$ percentage points more likely to be erased for layer4→head and $16.7$ percentage points more likely for layer3→head. This result shows that late learning alone is insufficient to explain the observed functional suffix dependence.

Table D2: Robustness and specificity of the learning-time–suffix-dependence relationship. Gaps are reported in percentage points.

| | Layer4→Head | Layer3→Head |
|---|---|---|
| Raw late–early gap (pp) | 35.3 | 43.1 |
| Final-margin matched gap (pp) | 26.8 | 34.9 |
| Std. learning-time coefficient | 0.414 | 0.447 |
| ΔCV-AUC | +0.0040 | +0.0058 |
| Permutation $p$-value | 0.002 | 0.002 |
| Late noisy–matched late clean (pp) | +21.4 | +16.7 |

Across final-margin matching, multivariate regression, cross-validation, within-run permutation testing, and matched late-clean comparisons, the same qualitative conclusion is preserved: later-learned incorrect labels exhibit stronger functional suffix dependence, and this relationship cannot be fully explained by final confidence, generic example difficulty, class identity, or run-level variation.

# E WITHIN-AUGMENTATION CONTROL: PAIRING-INDUCED LEARNING ORDER AND SEMANTIC PRESERVATION

The main experiments show that strong augmentation both enlarges learning-order differences among incorrect labels and preserves substantially more true-class structure. A natural concern is therefore that the observed association between learning order and semantic preservation may be explained entirely by augmentation strength: strong augmentation could independently delay noisy-label acquisition while also improving representation quality. To examine this possibility, we perform an additional controlled comparison entirely within the same strong-augmentation regime.

We reuse the transition-matched target-pairing construction introduced in Section 3.2. The matched-high, matched-low, and matched-random conditions contain exactly the same corrupted examples and the same multiset of assigned wrong targets. Consequently, they preserve the same noise rate, target-class frequencies, and source–target transition counts. Only the one-to-one assignment between individual examples and incorrect targets is changed. All three conditions are trained with the same strong-augmentation policy and identical optimization settings within each random seed.

In the primary setting, these three pairings induce the empirical learning-time ordering

$$L_{\text{high}} < L_{\text{low}} < L_{\text{random}}, \tag{24}$$

with matched-high errors learned $27.64$ epochs earlier on average than matched-random errors.

Importantly, we do not interpret these three conditions as a monotonic sweep over the scalar example–target compatibility score $S_i$. Although the matched-high assignment is learned substantially earlier than the matched-random assignment, matched-low occupies an intermediate position rather than being learned latest. This observation indicates that example–target compatibility alone is not a sufficient scalar determinant of learning time. Instead, changing the example–wrong-target assignment changes a broader relational configuration of the corrupted training set. Accordingly, we use the three conditions here as controlled *pairing-induced learning-order regimes*, rather than as evidence for a monotonic compatibility-to-learning-time mapping.

The purpose of this experiment is therefore different from testing whether $S_i$ monotonically determines $L_i$. We instead ask whether the learning-order differences that emerge from controlled target pairing are systematically associated with different semantic costs when augmentation strength and global corruption statistics are held fixed.

To compare semantic preservation independently of the total amount of memorized noise, we evaluate all three conditions at the same noise-fitting events

$$q \in \{0.50, 0.90, 0.95, 0.98\}.$$

Following Appendix D, we measure true-label 30-NN neighborhood purity over corrupted examples using the fixed non-augmented evaluation view. We report the layer4 representation, where the degradation of true-class structure caused by noisy-label memorization is most clearly visible.

Table E1: Within-augmentation true-label 30-NN neighborhood purity (%) at matched noise-fitting events. All three conditions use the same strong-augmentation regime, the same corrupted examples, the same wrong-target multiset, and the same source–target transition statistics. The three pairings induce the empirical learning-time ordering Matched-high $<$ Matched-low $<$ Matched-random. Random–High denotes the difference between the latest- and earliest-learned conditions. This comparison is used to test whether empirically induced learning order is associated with semantic preservation; it is not interpreted as a monotonic sweep over scalar target compatibility.

| $q$ | Matched-high | Matched-low | Matched-random | Random–High |
|---|---|---|---|---|
| 0.50 | 73.9 | 74.5 | 75.0 | +1.1 |
| 0.90 | 63.8 | 65.1 | 66.4 | +2.6 |
| 0.95 | 59.2 | 60.8 | 62.4 | +3.2 |
| 0.98 | 53.5 | 55.3 | 57.0 | +3.5 |

Table E1 reveals a consistent correspondence between the *empirically observed learning order* and semantic preservation within the same augmentation regime. Matched-high is learned earliest and

retains the weakest true-class neighborhood structure; matched-random is learned latest and retains the strongest; matched-low lies between the two on both quantities. Thus, although scalar target compatibility does not monotonically determine learning time, the learning-order differences induced by the controlled pairing intervention are systematically associated with different levels of semantic preservation.

This relationship becomes more pronounced as noisy-label fitting progresses. At $q = 0.50$, matched-random exceeds matched-high by only $1.1$ percentage points in true-label neighborhood purity. The difference increases to $2.6$ points at $q = 0.90$, $3.2$ points at $q = 0.95$, and $3.5$ points when $98\%$ of the incorrect labels have been fitted. Over the same interval, neighborhood purity decreases from $73.9\%$ to $53.5\%$ for matched-high, compared with $75.0\%$ to $57.0\%$ for matched-random. Hence, the condition in which incorrect labels are acquired earlier accumulates greater degradation of the underlying true-class geometry as memorization approaches completion.

This comparison is important because it does not rely on changing augmentation strength. Matched-high, matched-low, and matched-random use the same strong augmentation, the same corrupted examples, the same multiset of wrong targets, and the same source–target transition statistics. Augmentation strength therefore cannot by itself explain the ordering of semantic preservation across the three conditions. Together with the sample-level relationship between relational support and stable learning time reported in Figure 3(d), this result shows that controlled changes in example–wrong-target relations can induce systematic differences in when incorrect labels are acquired, and that the resulting learning-order differences are accompanied by corresponding differences in representational cost.

At the same time, this experiment should not be interpreted as a timing-only causal intervention. The transition-matched construction changes the example–wrong-target relation in order to induce different learning-order regimes, and the changed relation may itself directly influence representation learning. We therefore do not claim that learning time alone causally determines semantic damage. Our narrower conclusion is that, when augmentation strength, corrupted examples, wrong-target multisets, and global source–target corruption statistics are fixed, pairing-induced conditions that acquire incorrect labels earlier consistently retain less true-class structure at matched noise-fitting levels.

The pairing-based analysis therefore establishes a controlled association between learning order and semantic preservation, but does not isolate the temporal contribution itself. To address this remaining ambiguity, we next introduce a fixed-pair timing intervention that preserves each example–wrong-target relation while changing only the temporal placement of its erroneous supervision.

### E.1 Fixed-Pair Timing Intervention

The transition-matched pairing intervention changes which incorrect target is assigned to an individual example. Although this construction preserves the corrupted examples, noise rate, target frequencies, and source–target transition counts, it changes both the resulting learning order and the underlying example–target relation. We therefore design a second intervention that keeps the pair $(x_i, \tilde{y}_i)$ fixed and changes only the temporal placement of its erroneous supervision.

Let $w_i(t) \in \{0, 1\}$ denote whether the noisy-label loss of example $i$ is active at epoch $t$:

$$\mathcal{L}_i(t) = w_i(t)\,\mathrm{CE}\left(p_{\theta_t}(x_i), \tilde{y}_i\right).$$

For the early schedule,

$$w_i^{\mathrm{E}}(t) = \mathbf{1}[0 \le t < 50] + \mathbf{1}[100 \le t < 160],$$

whereas the delayed schedule is

$$w_i^{\mathrm{D}}(t) = \mathbf{1}[50 \le t < 100] + \mathbf{1}[100 \le t < 160].$$

Both schedules therefore contain exactly $110$ active supervision epochs per example.

To prevent systematic sample differences from confounding the timing comparison, corrupted examples are partitioned within each source–target transition into two balanced subsets $\mathcal{I}_A$ and $\mathcal{I}_B$. For each random seed, we train two paired runs. The first assigns the early schedule to $\mathcal{I}_A$ and the delayed schedule to $\mathcal{I}_B$; the second reverses these assignments. The paired runs share initialization,

mini-batch ordering, augmentation policy, and all optimization hyperparameters. Consequently, every corrupted example is observed once under each timing condition while retaining the same incorrect target.

To match the amount of memorized noise across the two timing conditions, we compute the noise-fitting rate separately for the early and delayed subsets,

$$q_s(t) = \frac{1}{|\mathcal{I}_s|} \sum_{i \in \mathcal{I}_s} \mathbf{1}[\hat{y}_i(t) = \tilde{y}_i], \qquad s \in \{\mathrm{E}, \mathrm{D}\},$$

and evaluate each condition at its own first checkpoint $\tau^s_{0.98} = \min\{t : q_s(t) \geq 0.98\}$. Thus, semantic comparisons are made at matched subgroup-specific noise-fitting levels rather than at a common training epoch.

For each example, semantic preservation is evaluated from the layer4 representation at the corresponding subgroup-specific $q = 0.98$ event. We first use the same true-label 30-NN purity metric as in Appendix D. We additionally define the true-class representation margin

$$M_i = h_i^\top \nu_{y_i} - \max_{c \neq y_i} h_i^\top \nu_c,$$

where $h_i$ is the normalized layer4 representation and $\nu_c$ is the normalized prototype of true class $c$. True labels are used only for post-hoc semantic evaluation and never for training.