Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Representational change under post-hoc edits mostly reflects relocation rather than destruction of information

Sajib Hossain

sajib.hossain03@northsouth.edu

North South University

Article

Keywords: representation learning, concept erasure, machine unlearning, continual learning, probing classifiers, representational drift

Posted Date: November 5th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10142965/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Representational change under post-hoc edits mostly reflects relocation rather than destruction of information

Sajib Hossain (ORCID: 0009-0005-6656-0128)[1*]

[1*]Department of Computer Science and Engineering, North South University, Dhaka, Bangladesh.

Corresponding author(s). E-mail(s): sajib.hossain03@northsouth.edu;

Abstract

Trained representations are edited by continual learning, model editing, and concept erasure, and a large change is commonly read as evidence that information was destroyed. We test this reading on testbeds with ground-truth factors (a β-VAE on dSprites; convolutional classifiers on Split-MNIST and CIFAR-10), separating information moved to new coordinates from information that is gone, using frozen versus fresh probes and latent-health metrics (effective rank, coordinate scale). Two linear edits, belief revision and LEACE erasure, share one dominant outcome: the original readout breaks, yet a fresh probe recovers the factor. Under belief revision at a healthy step size, permanent loss averages 1–2 percentage points over factors; the result transfers to real convolutional networks, although how much drift a cheap affine map reverses is substrate-dependent. Relocation is not universal: under linear erasure, nonlinear leak-back ranges from 0.54 to 1.00 over 32 cells, and scale loses up to 46% of its nonlinearly recoverable information at a healthy latent. In our experiments, total loss occurs only through representational collapse under an excessive optimizer step. Drift correlates only modestly with loss within a sweep and does not order outcomes across configurations; change magnitude is a poor proxy for loss.

Keywords: representation learning, concept erasure, machine unlearning, continual learning, probing classifiers, representational drift

1 Introduction

A large body of work modifies a trained representation *after* training: continual learning updates it on new tasks, model editing rewrites stored facts, concept erasure and debiasing remove a target attribute, and machine unlearning attempts to delete the influence of particular training data. Across all of these settings, the same diagnostic question recurs:

> When the representation changes a lot, has information been destroyed, or has it only been relocated to new coordinates?

In practice, this question is entangled with two confounds. First, "the readout broke" is taken to mean "the information is gone," even though a broken frozen probe is equally consistent with the information having moved to a new basis. Second, "the representation moved a lot" is taken to mean "a lot was lost," even though the magnitude of geometric change and the amount of information destroyed need not be related at all.

We disentangle these confounds in controlled testbeds where the ground truth is known: disentangled β-VAE latents on dSprites [1, 2], where each generative factor (shape, scale, x, y) has a measurable decodability, together with Split-MNIST and CIFAR-10 convolutional classifiers that test whether the conclusions survive a nonlinear encoder trained on real images. In both real-encoder settings, a fine-tuned encoder breaks the frozen old-task head, yet a freshly retrained head recovers it to within two points of, or above, the original accuracy (Sections 4.2–4.3). The practical stakes are concrete: if post-hoc edits are predominantly relocation, then cheap realignment maps fitted without replay may restore legacy downstream heads; and if total loss arises only through collapse (vanishing latent variance and rank), then monitoring the magnitude of change is misleading on both counts, and one should monitor latent health and fresh recoverability instead.

Our contribution is characterization and diagnostics rather than a new algorithm:

1. A dual-probe diagnostic protocol that distinguishes relocation from destruction, combining frozen versus fresh-from-scratch probes, replay-free affine realignment, and latent-health metrics.
2. A cross-mechanism result: for both belief-revision drift and LEACE concept erasure [3], relocation is the dominant outcome across three substrates—with two caveats. The *linear invertibility* of a drift is substrate-dependent even where its information recoverability is not, and linear erasure leaves a factor-dependent residue of partial destruction at a healthy latent (Section 4.7).
3. A collapse-not-drift boundary for *total* loss: in our experiments, the only route to total, irreversible loss is representational collapse driven by optimizer step size; neither contradiction load nor drift magnitude produces it. Drift is a poor proxy for loss: it correlates only modestly with loss within a sweep (Spearman $\rho = 0.45$–0.54) and fails across configurations, where a small drift can come with near-total loss (the $\times 16$ level on dSprites) and a large drift with almost none (the Split-MNIST fine-tune), each measured in the units of its own feature layer.
4. Two negative results that bound the claim: an a-priori geometric predictor of nonlinear leak-back fails (Spearman $\rho = 0.21$, $p = 0.25$), and disentanglement pressure

β leaves mean leak-back unchanged (0.82–0.86) while redistributing it across factors (scale rises from 0.55 at $\beta = 1$ to 0.75–0.79 at $\beta \geq 4$; shape falls from 1.00 to 0.83).

The relocation *phenomenon* has been observed in adjacent literatures (Section 2); what we add is the controlled, cross-mechanism characterization, the magnitude-versus-loss frontier, and the explicit negative results—not the discovery of relocation itself. The paper makes a single claim: the magnitude of representational change does not measure information loss. We state it in its defensible form: relocation is the dominant outcome of small linear edits on a healthy representation, partial destruction occurs and is factor-dependent, and change magnitude remains a poor proxy for loss. We test it on two distinct editing families that share one boundary for total loss, representational collapse. The mechanisms serve that argument; they are not independent case studies.

1.1 Claims and scope

This is an analysis and characterization paper rather than a methods paper. It uses existing techniques—linear and nonlinear probing, disentangled VAEs, closed-form concept erasure, and affine or orthogonal realignment—to clarify what post-hoc representation edits do and do not destroy, rather than to propose a new method. Each claim maps to a specific controlled experiment; Table 8, at the end of Section 4, summarizes every claim together with the result that supports it.

We do *not* claim the relocation phenomenon, the probe diagnostic, the realignment map, or LEACE as novel; all are prior art (Section 2). We claim only the controlled cross-mechanism characterization with ground-truth factors, the collapse-not-magnitude frontier, and the two negative results. The claims are deliberately bounded: each is tied to the experiment that supports it, and the scope of every claim is stated explicitly.

2 Related work

Continual learning and forgetting.

A large family of methods limits forgetting on past tasks through weight regularization (elastic weight consolidation), functional and feature distillation [4], exemplar replay with incremental representation learning [5], and orthogonal-gradient updates. We do not propose a competing method; we characterize what a representation edit *means*, geometrically and functionally, in a probe-grounded sense.

Drift compensation.

Semantic Drift Compensation [6], Backward Feature Projection [7], and Learnable Drift Compensation [8] *correct* representations after drift. We ask whether correction is *necessary* for information preservation, and show that under healthy training the information survives the drift on every substrate we test, while how much of the drift a replay-free linear map can invert varies from 91% (Split-MNIST) to 13% (CIFAR-10).

Probing and representational forgetting.

Davari et al. [9] probe forgetting; we extend the logic with *fresh* probes that re-ask whether information still exists in new coordinates, rather than whether an old readout still works.

Concept erasure.

INLP [10], RLACE [11], and LEACE [3] remove *linear* predictability of a concept; LEACE guarantees linear guardedness in closed form and explicitly conjectures that nondestructive *nonlinear* erasure is intractable "unless the data-generating process is known." Kernelized erasure [12] shows that nonlinear protection does not transfer across adversaries. The observation that erasure relocates rather than destroys has surfaced in other substrates: diffusion concept-editing [13], NLU debiasing [14], and LLM unlearning [15]. We add a controlled disentangled-factor characterization in which the data-generating process *is* known, and we report as negative results that a geometric a-priori predictor of leak-back does not work and that disentanglement changes which factor is partially destroyed rather than how much leaks back on average.

Stability–plasticity and collapse.

Loss of plasticity [16] and the stability gap [17] are known degradation phenomena under repeated training. We connect "apparent irreversibility" at high learning rate directly to variance and rank collapse, rather than to structured erasure.

What we add.

None of the prior work jointly reports (i) frozen versus fresh probe separation, (ii) replay-free affine recovery rates, (iii) latent-health versus information-loss phase diagrams, (iv) the *same* relocation conclusion across two distinct linear mechanisms, and (v) a negative result showing that change magnitude fails as a destruction proxy under controlled disentangled probing. Prior work established that information can remain recoverable after editing; our contribution is to isolate *when* apparent destruction instead corresponds to representational collapse, using frozen/fresh probes and latent-health diagnostics across two editing families. The aim is neither a new editing method nor the observation that representations can retain recoverable information. Relocation, probing, affine recovery, and LEACE behavior each have prior-art adjacency on their own; their controlled joint characterization, and the collapse-versus-drift separation, do not.

3 Methods

3.1 Notation and diagnostic quantities

Let $f_\theta : \mathcal{X} \to \mathbb{R}^d$ be the encoder ($d = 16$ for the β-VAE), producing latents $z = f_\theta(x)$, and let $y \in \{1, \ldots, K\}$ be a ground-truth factor label. A post-hoc edit maps $f_\theta \to f_{\theta'}$; we write the pre- and post-edit latents as $z_i^{\mathrm{b}} = f_\theta(x_i)$, $z_i^{\mathrm{a}} = f_{\theta'}(x_i)$, with matrices

$Z^{\mathrm{b}}, Z^{\mathrm{a}} \in \mathbb{R}^{n\times d}$. A probe $g : \mathbb{R}^d \to \{1, \ldots, K\}$ has accuracy

$$A(g; Z, y) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}[\, g(z_i) = y_i \,]. \tag{1}$$

Let g^{fr} be a *frozen* probe trained on Z^{b} and g^{fresh} a *fresh* probe retrained on Z^{a}, giving base, broken, and fresh accuracies $a_{\mathrm{base}} = A(g^{\mathrm{fr}}; Z^{\mathrm{b}}, y)$, $a_{\mathrm{brk}} = A(g^{\mathrm{fr}}; Z^{\mathrm{a}}, y)$, $a_{\mathrm{fresh}} = A(g^{\mathrm{fresh}}; Z^{\mathrm{a}}, y)$. The two functional quantities at the center of our analysis are

$$\Delta_{\mathrm{fr}} = \underbrace{a_{\mathrm{base}} - a_{\mathrm{brk}}}_{\text{frozen drop (readout broke)}}, \qquad L_{\mathrm{perm}} = \underbrace{a_{\mathrm{base}} - a_{\mathrm{fresh}}}_{\text{permanent loss (information gone)}}. \tag{2}$$

We call the outcome *relocation* when $\Delta_{\mathrm{fr}} > 0$ but $L_{\mathrm{perm}} \approx 0$, and *destruction* when $L_{\mathrm{perm}} \gg 0$. Both conditions are used throughout: a broken frozen readout ($\Delta_{\mathrm{fr}} > 0$) is the precondition that makes the question interesting, and a near-zero permanent loss is what then distinguishes relocation from destruction. Because raw accuracy has a different range for each factor (the chance level c ranges from 1/32 to 1/3 across the dSprites factors), we also report the *chance-normalized* permanent loss

$$\tilde{L}_{\mathrm{perm}} = \frac{a_{\mathrm{base}} - a_{\mathrm{fresh}}}{a_{\mathrm{base}} - c}, \tag{3}$$

the fraction of the decodable range that a fresh probe no longer recovers; $\tilde{L}_{\mathrm{perm}} = 1$ means the factor has fallen to chance. This is the same normalization as the leak-back ratio of Eq. (8) below, with $1 - \ell$ playing the role of $\tilde{L}_{\mathrm{perm}}$.

Realignment.

We fit an affine map $M \in \mathbb{R}^{(d+1)\times d}$ from post- to pre-edit latents, and an orthogonal (Procrustes) variant R:

$$M = \arg\min_{M} \left\| [\, Z^{\mathrm{a}} \; \mathbf{1} \,] M - Z^{\mathrm{b}} \right\|_F^2, \quad R = \arg\min_{R^\top R = I} \left\| (Z^{\mathrm{a}} - \bar{z}^{\mathrm{a}})R - (Z^{\mathrm{b}} - \bar{z}^{\mathrm{b}}) \right\|_F = UV^\top, \tag{4}$$

where $U\Lambda V^\top$ is the SVD of $(Z^{\mathrm{a}} - \bar{z}^{\mathrm{a}})^\top (Z^{\mathrm{b}} - \bar{z}^{\mathrm{b}})$; a *replay-free* map fits M on current-task data only. With realigned accuracy a_{re} and realigned latents $\hat{Z}$, recovery and map fidelity are

$$\rho_{\mathrm{rec}} = \frac{a_{\mathrm{re}} - a_{\mathrm{brk}}}{a_{\mathrm{base}} - a_{\mathrm{brk}}}, \qquad R^2 = 1 - \frac{\|\hat{Z} - Z^{\mathrm{b}}\|_F^2}{\|Z^{\mathrm{b}} - \bar{z}^{\mathrm{b}}\|_F^2}. \tag{5}$$

In words, ρ_{rec} is the fraction of the broken frozen-probe gap that the learned map restores (1 means the old readout is fully recovered, 0 that it is not helped at all), and R^2 measures how closely the realigned latents reproduce the originals.

Latent health.

For latent covariance $\Sigma = \mathrm{Cov}(Z)$ with eigenvalues $\lambda_1, \ldots, \lambda_d \geq 0$, the effective rank (participation ratio), mean coordinate scale, and drift are

$$r(\Sigma) = \frac{\left(\sum_i \lambda_i\right)^2}{\sum_i \lambda_i^2} = \frac{\mathrm{Tr}(\Sigma)^2}{\mathrm{Tr}(\Sigma^2)} \in [1, d], \quad s(Z) = \frac{1}{d}\sum_{j=1}^{d} \mathrm{std}(Z_{\cdot j}), \quad \delta = \frac{1}{n}\sum_{i=1}^{n} \|z_i^{\mathrm{a}} - z_i^{\mathrm{b}}\|_2. \tag{6}$$

Collapse is a simultaneous drop in r, s, and a_{fresh}: the latent contracts toward a few low-variance directions just as the factor becomes undecodable. We distinguish this from relocation (large drift δ with r and s intact).

Regime rule.

Tables 4 and 5 assign each configuration a regime label by one fixed rule, applied to the ratios of r and s to their pre-edit values and to the raw permanent loss averaged over factors. A latent is *healthy* if $r/r_{\mathrm{base}} > 0.6$ and $s/s_{\mathrm{base}} > 0.4$; information is *destroyed* if $L_{\mathrm{perm}} > 0.10$. A configuration is labeled *relocation* if it is not destroyed, *collapse* if it is destroyed and not healthy, and *structured destruction* if it is destroyed while healthy. The rule does not test $\Delta_{\mathrm{fr}} > 0$ separately, because the frozen readout broke in every configuration where it was measured: the mean frozen drop is 19–42% across the levels of the learning-rate sweep (Table 3) and 12–27% across the seeds of the rounds sweep (derived from the per-factor broken accuracies in that run's log), and the smallest mean frozen drop reported anywhere in the paper is 5.6%, at overlap $b = 0$ (Table 1), where the update subspace contains no semantic dimension; even there the smallest single run breaks the readout by 2.9%. The collapse-check run did not log the frozen probe, but it repeats the learning-rate levels of Table 3 on the same encoder. A configuration with $L_{\mathrm{perm}} \leq 0.10$ and an intact frozen readout would be an unchanged representation rather than relocation, and none occurred where the frozen probe was measured. The two thresholds are heuristics fixed in the analysis code rather than calibrated quantities; every table therefore reports the underlying ratios so that a reader can apply a different rule, and Section 4.5 discusses the one row (learning-rate scale $\times 4$) whose label depends on the scale threshold. Two further cautions apply. First, s is not invariant to a uniform rescaling of the latent, whereas r and the random-forest probes are; a drop in s alone is therefore never read as collapse unless fresh decodability falls with it, which a uniform rescaling cannot produce. Second, both statistics are reported as ratios to the pre-edit baseline of the same layer, because the absolute participation ratio depends on the task and layer; absolute values are given alongside the ratios in Sections 3.2, 4.3, and 4.5.

Concept erasure (Mechanism 2).

Here LEACE acts on the latent itself to erase a target factor. Let X be the centered latent features (the matrix Z centered), Y the one-hot encoding of the factor label y, and Σ_{XY} their cross-covariance. The LEACE eraser [3] is the affine map

$$r_{\mathrm{LEACE}}(x) = x - \Sigma^{1/2} P_{\Sigma^{-1/2}\Sigma_{XY}} \Sigma^{-1/2} (x - \mu), \tag{7}$$

where $\Sigma = \mathrm{Cov}(X)$ and P_W projects orthogonally onto the column space of W; the map equalizes class-conditional means, so $\mathrm{Cov}(r_{\mathrm{LEACE}}(X), Y) = 0$ and no linear probe can recover the factor. With a fresh probe of accuracy a^{er} on the erased latent and chance level c, the leak-back of a probe family is

$$\ell = \frac{a^{\mathrm{er}} - c}{a_{\mathrm{base}} - c}, \tag{8}$$

the fraction of the concept surviving linear erasure (with $\ell \in [0, 1]$ whenever $c \le a^{\mathrm{er}} \le a_{\mathrm{base}}$): $\ell_{\mathrm{lin}} \approx 0$ by construction, while ℓ_{nl} (a fresh *nonlinear* probe) measures nonlinear leak-back.

3.2 dSprites β-VAE and preprocessing

We train a convolutional β-VAE on dSprites with $\beta = 1$, latent dimension 16, and 200k samples (seed 42). One implementation detail proved consequential: dSprites renders binary $\{0, 1\}$ images, and scaling pixels by $1/255$ drives the posterior toward total collapse (latent std $\to 0$). Our decodable latents depend on treating the inputs as binary, and we report this as a cautionary reproducibility note. We study the four decodable factors: shape (3 values), scale (6), x (32), and y (32); orientation is near-undecodable in our latents and is excluded. Baseline fresh random-forest probe accuracies are shape 0.91, scale 0.62, x 0.32, and y 0.33, and the latent has effective rank 11.4/16 with mean per-dimension std 0.73.

3.3 Mechanism 1: iterated belief revision

Beliefs are shape→color associations held by a small belief module that reads the VAE latent and predicts the color bound to each shape (Appendix A); the initial beliefs assign one color to each of the three shapes. A *contradiction* is a training signal that asserts a different color for one shape than the one the module currently holds (for example, square→blue in place of square→red): the revision targets are the current shape-to-color map with that one entry replaced, and the module and encoder are trained by cross-entropy on the predicted color to absorb the conflicting association. Revising the beliefs to absorb a contradiction is the edit whose effect on the latent we measure. Each *round* applies one contradiction for 10 epochs of projection-based revision: the gradient is confined to a 2-d belief subspace. We use a cumulative color map across rounds so the model is not implicitly trained to unlearn prior contradictions. In overlap sweeps, the belief (update) subspace is constructed to contain exactly $b \in \{0, 1, 2\}$ true semantic latent dimensions.

3.4 Mechanism 2: linear concept erasure (LEACE)

We apply LEACE [3] (closed-form least-squares linear erasure) to remove a factor from the 16-d latent via the eraser r_{LEACE} of Section 3.1, which equalizes class-conditional means so that no linear classifier can recover the concept. Each factor is coarsened to $K = 3$ quantile classes so the removed subspace is small (≤ 2 dimensions) and cannot trivially collapse the latent. For scale, x, and y this yields three balanced

classes (majority-class rate 0.34–0.35) and LEACE removes two dimensions; for shape, which takes only three values, the quantile cut merges two of them, so the shape task is in effect two-class (majority-class rate 0.66) and LEACE removes one dimension. Consequently the same factor names denote different prediction tasks in the two mechanisms: Mechanism 1 probes the uncoarsened factors (3, 6, 32, and 32 classes), whereas Mechanism 2 probes the coarsened versions, and the two are compared only through the chance-normalized quantities $\tilde{L}_{\text{perm}}$ and ℓ (for Mechanism 2, c in Eq. (8) is the majority-class rate of the coarsened label). Leak-back is clipped to $[0, 1]$; where a post-erasure probe slightly outperforms the pre-erasure probe (four shape cells, raw ratio 1.02–1.09), the reported value is 1.00. Keeping the removed subspace small is a control against trivial collapse, but it also means that the erasure grid does not contain the large-subspace edits most likely to destroy information at a healthy latent (Section 6). We sweep $\beta \in \{1, 2, 4, 8\}\times$ factor $\times$ 2 seeds (32 cells), where each seed fixes the 50/50 split of the evaluation latents, the eraser fitted on its training half, and the random state of the probes; the β values here are the same VAE hyperparameter as in the robustness arm of Section 4.9, which covers $\beta \in \{1, 2, 4\}$ for Mechanism 1 because the belief module was trained only at those three values.

3.5 Dual-probe diagnostic and realignment

We instantiate the frozen and fresh probes of Section 3.1 with both linear (logistic regression) and nonlinear (random-forest) families, and fit the realignment maps M (affine) and R (orthogonal Procrustes), including the replay-free variant fit on current-task data only. The diagnostic quantities Δ_{fr}, L_{perm}, ρ_{rec}, and R^2 are exactly as defined there; a broken frozen readout ($\Delta_{\text{fr}} > 0$) together with $L_{\text{perm}} \approx 0$ indicates relocation, and $\tilde{L}_{\text{perm}}$ measures how far an outcome departs from it. Probe accuracy reflects what a given probe family can *decode* rather than information-theoretic presence, and it is sensitive to probe capacity [18, 19]; we therefore report both linear and nonlinear families and treat a freshly retrained probe, not a frozen readout, as the test of recoverability.

3.6 Latent-health metrics

We monitor the effective rank $r(\Sigma)$ and mean coordinate scale $s(Z)$ defined in Section 3.1. Collapse is operationalized as a simultaneous drop in r, s, and fresh decodability a_{fresh}, not as high change magnitude alone.

3.7 Experimental sweeps and real-encoder tasks

The real-encoder experiments use a two-task split. On Split-MNIST, Task A is classification of the digits 0–4 and Task B of the digits 5–9; on CIFAR-10, Task A is the classes 0–4 (airplane, automobile, bird, cat, deer) and Task B the classes 5–9 (dog, frog, horse, ship, truck). In each case a linear head for Task A is trained on the pre-edit features, the whole encoder is then fine-tuned on Task B with a fresh head, and that fine-tuning is the edit under study: the frozen probe is the Task-A head, and the fresh probe is a Task-A head retrained on the post-edit features. On dSprites, the overlap sweep varies $b \in \{0, 1, 2\}$ at the base learning rate; the learning-rate sweep scales

the revision learning rate by $\{1, 2, 4, 8, 16, 32\}$ at maximal overlap; the collapse check repeats $\{1, 4, 8, 16, 32\}$ as a separate run that also logs latent health; and the rounds sweep accumulates 6/24/48 contradiction rounds at the healthy step size. The number of seeds differs by sweep: five partition seeds for the overlap sweep, three for the learning-rate sweep, two for the collapse check and the rounds sweep, and two seeds for each erasure cell (each setting the split and the probe's random state, Section 3.4); the belief-revision probes use a fixed random state. Wherever a mean±std is reported, the std is the population standard deviation over those seeds, as logged. Full architectures, hyperparameters, and training configurations for all substrates are given in Appendix A.

4 Results

The results develop a single argument rather than two parallel studies. Sections 4.1–4.2 and 4.7 test the relocation claim with two mechanistically distinct linear edits, belief-revision drift and LEACE erasure; Sections 4.4–4.6 locate the boundary at which relocation gives way to total loss (representational collapse); and Section 4.9 checks that the claim is not an artifact of a single encoder initialization or disentanglement level. Belief revision and concept erasure are different operations. The narrower point is that both, as linear post-hoc edits on a healthy representation, keep most of the information recoverable while moving it to new coordinates; Section 4.7 quantifies the exception. Table 8 at the end of this section summarizes the claims and the evidence for each.

4.1 Belief revision: subspace overlap drives drift; frozen probes break; fresh probes recover

Across seeds, the number of semantic dimensions in the update subspace predicts geometric drift (Spearman $\rho = 0.85$, $p = 6 \times 10^{-5}$) and frozen-probe drop ($\rho = 0.83$); drift in turn predicts frozen-drop ($\rho = 0.89$). Table 1 reports the per-overlap means. Permanent loss also rises with overlap, from 0.4% to 1.8% raw ($\rho = 0.60$, $p = 0.017$, $n = 15$ runs), and the rise is carried almost entirely by the scale factor: its raw loss reaches 6.2% at $b = 2$, which is 13.6% of its decodable range ($\rho = 0.85$ for scale alone), while shape, x, and y stay within about 1.5 points of baseline at every level with no trend in overlap. The effect is small next to the frozen drop of 19% at $b = 2$, but it is a real, factor-dependent residue of destruction and anticipates the pattern of Section 4.7.

At overlap = 2 and the base learning rate (the $\times 1$ level of Table 3 and the 6-round row of Table 5, five seeds in all), a 272-parameter affine realignment recovers 85–90% of the frozen-probe drop across the five seeds (run means 86% and 87%; 93–96% for shape), and fresh probes remain within about 2% of baseline on average. Most of the factor information has moved to new coordinates rather than being lost; the residue is the scale loss noted above (Figure 1).

Table 1 Belief revision on dSprites. As more true semantic dimensions b enter the update subspace, latent drift and the frozen-probe drop grow, while permanent loss (the fresh probe's residual failure) stays small but rises with overlap. Mean±std over five partition seeds; the last column is the chance-normalized permanent loss of Eq. (3) averaged over the four decodable factors.

Overlap b	Drift δ	Frozen drop Δ_{fr}	Permanent loss L_{perm}	Normalized $\tilde{L}_{\text{perm}}$
0	0.025 ± 0.012	$5.6 \pm 2.7\%$	$0.4 \pm 0.1\%$	0.9%
1	1.563 ± 0.743	$13.4 \pm 4.7\%$	$0.6 \pm 0.6\%$	0.8%
2	2.767 ± 1.079	$19.0 \pm 2.6\%$	$1.8 \pm 0.7\%$	4.4%

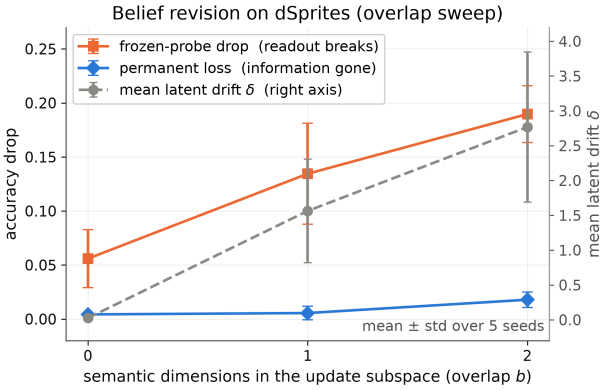


Fig. 1 Belief revision on dSprites. As more semantic dimensions enter the update subspace, the frozen-probe drop Δ_{fr} (orange) grows and the mean latent drift δ (gray dashed line, right axis) grows with it, but the permanent loss L_{perm} (blue; information unrecoverable by a fresh probe) stays small, rising from 0.4% to 1.8% with overlap in a rise carried by the scale factor (Section 4.1). Points are means and error bars are ±1 standard deviation over the five partition seeds. The readout breaks, but most of the information remains.

4.2 Belief-revision relocation transfers to a real CNN (Split-MNIST)

As Table 2 shows, the drift is an affine reparameterization rather than an orthogonal rotation, which demonstrates that relocation is not specific to disentangled VAEs. The drift is large ($\delta = 30.1$, in the units of the 128-dimensional feature layer; drifts measured on different substrates are not compared numerically), yet a retrained head recovers 0.974 against the original 0.990: a large change with almost no loss.

4.3 A third substrate (CIFAR-10): relocation generalizes, linear invertibility weakens

We repeat the real-encoder protocol on CIFAR-10 (a 5-vs-5 class split with a deeper 3-channel CNN). Fine-tuning on Task B breaks the frozen Task-A head from 0.712 to 0.518 (-19 points; feature drift $\delta = 10.4$). Under our diagnostic, this is once more relocation rather than destruction: a retrained head recovers to 0.748, above the original

Table 2 Belief-revision protocol on a real CNN (Split-MNIST). Fine-tuning on Task B breaks the frozen Task-A head, yet a replay-free affine map recovers most of the drop and a retrained head restores accuracy to within two points of the original. Drift is δ of Eq. (6), the mean Euclidean (L_2) displacement of a sample's feature vector. "Same distribution" means the realignment map is fit on pre- and post-edit features of Task-A data, an oracle that requires old data; "replay-free" fits it on Task-B features only. High affine ($R^2 = 0.97$) but low orthogonal (0.29) map fidelity indicates the drift is affine, not a pure rotation. Single training run.

Metric	Value
Frozen Task-A accuracy (pre-edit)	0.990
Frozen Task-A accuracy after Task-B fine-tune	0.753
Feature drift δ (mean per-sample L_2 displacement)	30.1
Affine realign, same distribution	0.980 (96% recovery)
Affine realign, replay-free (Task-B features only)	0.968 (91% recovery)
Orthogonal realign, same distribution / replay-free	0.946 (81%) / 0.879 (53%)
Retrained Task-A head (upper bound)	0.974
Map R^2, same distribution (affine / orthogonal)	0.97 / 0.29
Map R^2, replay-free (affine / orthogonal)	0.90 / 0.14

0.712, so the information is fully present, and the features do not collapse. In absolute terms the effective rank of the 128-dimensional feature layer rises from 2.69 to 2.87 (107% of baseline) and the mean coordinate std from 0.44 to 0.59 (135%). The absolute participation ratio is low in both states because a five-class cross-entropy head concentrates feature variance in a few directions; health is therefore judged relative to the pre-edit baseline of the same layer, and we report absolute values so that substrates can be compared (the dSprites latent has effective rank 11.4 of 16). A participation ratio this low cannot fall far, so on CIFAR-10 the health statistics are weak evidence on their own, and the relocation verdict rests on the retrained head. The *linear* invertibility of the drift, however, is markedly weaker than on Split-MNIST. A replay-free affine map recovers only 13% of the frozen-head drop (47% in the same-distribution case), with map R^2 of 0.80–0.84 against 0.97 on MNIST. On a deeper encoder over natural images, the drift carries a larger nonlinear component, so a linear map undoes less of it, even though the information itself remains fully recoverable. This pulls apart two properties that the dSprites and MNIST results had run together: the information stayed recoverable on every substrate, but a linear map could undo the drift only on Split-MNIST (Figure 2).

4.4 An apparent destruction frontier under learning-rate pressure

Sweeping the revision learning-rate scale at fixed maximal overlap, realignment recovery falls from 87% to 0% and permanent loss rises from 1% to 42% (Spearman $|\rho| \approx 0.95$; Table 3). On its face, this looks like adaptation pressure destroying information; Section 4.5 shows that it is collapse.

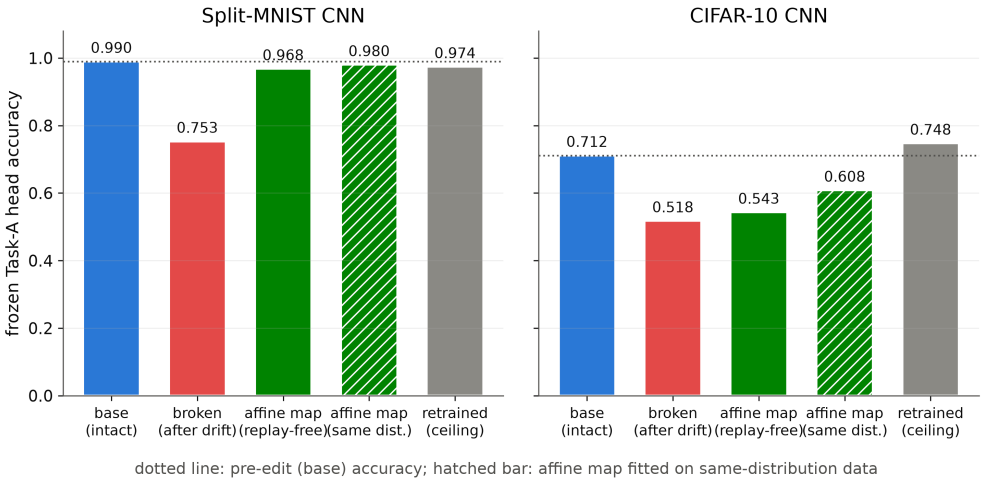


Fig. 2 Real-encoder relocation on two substrates. In both Split-MNIST and CIFAR-10, fine-tuning on a new task breaks the frozen old-task head, yet a retrained head recovers to within two points of the original accuracy on Split-MNIST (0.974 against 0.990) and above it on CIFAR-10 (0.748 against 0.712): the information is present, not destroyed. The difference lies in the replay-free *affine* recovery (solid green; the hatched green bar is the same-distribution map and the dotted line the pre-edit accuracy), which reaches the retrained ceiling on Split-MNIST but barely clears the broken baseline on CIFAR-10. A fresh head recovers the information on both substrates; a linear map restores the old readout only on Split-MNIST. One training run per substrate (Table 2).

4.5 The destruction is representational collapse, not structured loss

Table 4 makes the mechanism explicit. Under the rule of Section 3.1, every row of this sweep is either relocation or collapse; no row is labeled as destroyed at a healthy latent. In absolute terms, effective rank falls from 11.4 (of 16) to 10.3, 7.2, 2.2, 2.3, and 0.0 and mean coordinate std from 0.730 to 0.69, 0.14, 0.06, 0.04, and 0.00 across $\times 1$ to $\times 32$. The $\times 4$ row is the boundary case and deserves a direct statement. Its mean effective rank, 63% of baseline (54% and 71% in the two seeds; absolute 7.2 against 11.4), passes the rank criterion, so its label as collapse rests on the coordinate scale, which has fallen to 19% of baseline (absolute mean std 0.135 against 0.730). A uniform rescaling of the latent would lower s without affecting the random-forest probes, so the scale drop alone would not justify the label. What rules out a mere rescaling is that fresh decodability falls at the same time, by 51% of the decodable range after chance normalization (per factor: shape 71%, scale 51%, x 44%, y 37%), which a rescaling cannot produce. This argument excludes a benign explanation of the scale drop; it does not by itself prove collapse, because a fall in decodability is common to collapse and structured destruction, and the label remains threshold-sensitive: seed 0 (rank 71%, std 27%, raw loss 21%) would be labeled structured destruction at a std threshold of 25% rather than 40%, whereas seed 1 (rank 54%, std 10%) is collapse under any rule. We read the row as the onset of collapse; a reader who weights rank alone would instead call it a healthy latent that lost half of its information, and we flag that dependence rather than hide it. Beyond $\times 4$ the picture is unambiguous: per-seed

Table 3 Apparent destruction frontier under learning-rate pressure (dSprites, maximal overlap). As the revision learning-rate scale rises, realignment recovery falls and permanent loss climbs, while drift itself is non-monotone. Frontier run: mean (±std where shown) over three partition seeds per level. Table 4 comes from a separate collapse-check run with two seeds and without the ×2 level, which is why its loss values differ slightly from these at ×8 and ×16. Section 4.5 shows the cause is collapse, not structured loss.

LR×	drift δ	frozen drop	recovery	permanent loss
1	3.5	22%	$87 \pm 2\%$	1.1%
2	2.0	19%	$83 \pm 2\%$	1.5%
4	17.7	42%	$31 \pm 4\%$	22.0%
8	20.7	42%	$8 \pm 2\%$	33.3%
16	5.5	42%	-1%	41.6%
32	12.4	42%	0%	41.7%

Table 4 The apparent frontier is representational collapse (dSprites). Over the learning-rate sweep, permanent loss rises only as effective rank, coordinate scale (std), and fresh decodability fall together. Collapse-check run: two partition seeds; each health entry is the mean with the per-seed range in parentheses. It is a separate run from the frontier sweep of Table 3 (three seeds), which is why the raw loss differs slightly at ×8 and ×16 (29% and 35% here against 33% and 42% there). "raw loss" is the permanent loss averaged over factors and "norm. loss" its chance-normalized version (Eq. (3)); both use this run's own per-factor baselines (shape 0.923, scale 0.622, x 0.336, y 0.353; mean 0.559). Regime labels follow the rule of Section 3.1 (healthy if rank $>$ 60% and std $>$ 40% of base; destroyed if raw loss $>$ 10%). At ×4 the rank criterion passes and the std criterion fails, so the label rests on std and is threshold-sensitive (seed 0, with rank 71%, std 27%, and raw loss 21%, would be labeled structured destruction at a std threshold of 25%); we mark it as the boundary of collapse. At ×8 and ×16 one seed is fully collapsed (rank 0) and the other only partially (rank about 40%), which the means conceal.

LR×	rank (% base)	std (% base)	fresh	raw loss	norm. loss	regime
1	91 (90–92)	94 (94–95)	0.548	1%	3%	relocation
4	63 (54–71)	19 (10–27)	0.332	23%	51%	collapse (boundary)
8	20 (0–40)	8 (0–16)	0.265	29%	67%	collapse
16	20 (0–41)	6 (0–12)	0.213	35%	81%	collapse
32	0 (0–0)	0 (0–0)	0.143	42%	100%	collapse

effective rank is 0–41% of baseline and the normalized loss reaches 67–100%. Chance normalization also changes how the raw numbers should be read. The "42% loss" at ×32 is total destruction (every factor at chance), and the "1%" at ×1 is a loss of about 3% of the decodable range (1.4% and 4.2% in the two seeds). Drift magnitude is a poor guide to this destruction. Within the learning-rate sweep, drift and permanent loss are only modestly correlated (Spearman $\rho = 0.45$, $p = 0.06$ over the 18 runs of Table 3; $\rho = 0.49$, $p = 0.33$ over its six level means, a test with no power at $n = 6$), and the ordering fails where it matters: at ×16 the drift is only 5.5, less than at ×4 (17.7) or

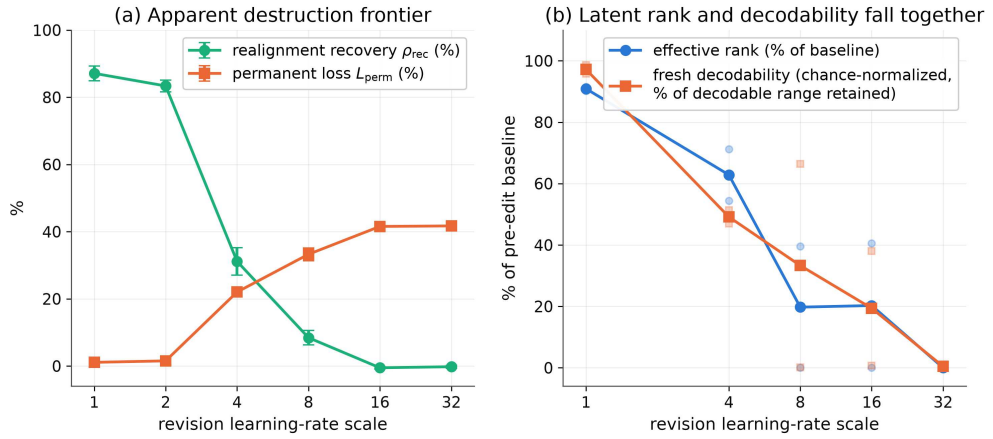


Fig. 3 The apparent destruction frontier is representational collapse. (a) As adaptation pressure (revision learning-rate scale, ×1 to ×32) rises, realignment recovery ρ_{rec} falls and permanent loss L_{perm} rises (frontier run, three seeds per level). (b) Over the collapse-check run (two seeds per level, faint points), effective rank $r(\Sigma)$ as a percentage of its pre-edit baseline and fresh decodability as the chance-normalized fraction of the decodable range that a fresh probe still recovers ($1-\tilde{L}_{\text{perm}}$ averaged over factors, the complement of the "norm. loss" column of Table 4, so that total collapse reads 0%) fall together: information is lost when the latent collapses, not in proportion to drift magnitude δ (for example, ×16 has small drift yet nearly total loss). The permanent loss in (a) is the raw factor average, of which 42% is total. Absolute values are in Tables 3 and 4.

×8 (20.7), yet the loss is nearly total. Within the overlap sweep of Section 4.1, where every run stays healthy, drift and loss are correlated ($\rho = 0.54$, $p = 0.04$, $n = 15$), but that correlation is carried by the scale residue ($\rho = 0.75$ for scale) and spans a loss range of 0.4–1.8 points. Drift is therefore informative only within a fixed sweep and at the level of small residues; across configurations it neither identifies collapse nor rules it out. Figure 3 shows the recovery collapse (a) coinciding with the death of latent rank and decodability (b).

4.6 Accumulated contradiction at a healthy step size does not destroy

An 8× increase in contradiction exposure produces no cliff in fresh decodability (Table 5, Figure 4); it is the optimization step size, not the contradiction load, that drives destruction. As in Section 4.1, the small residue that remains sits mostly on the scale factor (raw loss 0.7–6.3 points across the six runs, up to 14% of its decodable range at 48 rounds), and the chance-normalized loss averaged over factors stays within 1–6% in every run.

Table 5 Accumulated contradiction at a healthy step size (dSprites). An 8× increase in contradiction rounds (6 → 48) leaves latent health, fresh decodability, and recovery essentially flat: contradiction load alone does not destroy information. Rounds run: two partition seeds; ranges span the two seeds, "raw loss" is the permanent loss averaged over factors, and every row is labeled relocation by the rule of Section 3.1.

rounds	rank (% base)	std (% base)	fresh	raw loss	recovery
6	90–92%	93–95%	0.540–0.552	1–2%	86% (both seeds)
24	90–94%	93–95%	0.546–0.552	1%	74–75%
48	86–89%	91–97%	0.538–0.540	2%	77–81%

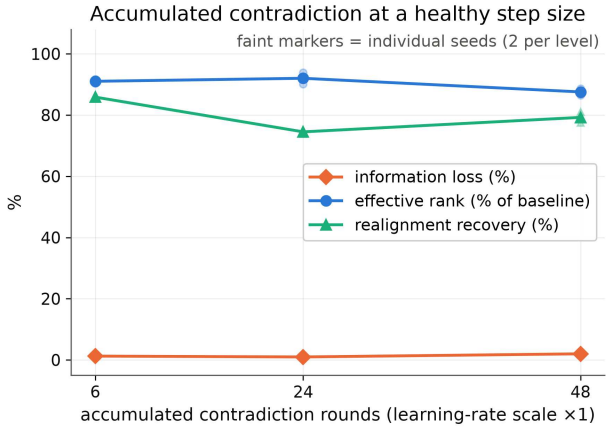


Fig. 4 Accumulated contradiction at a healthy step size does not destroy. Across 6 → 48 rounds, information loss stays at 1–2 percentage points (raw, averaged over factors; most of it sits on the scale factor, Section 4.6) while effective rank and realignment recovery remain high, with no decline toward a cliff. Lines are means over the two partition seeds; faint markers are the individual seeds.

4.7 Linear concept erasure: relocation dominates, with factor-dependent partial loss

Applying LEACE to erase a factor across 32 cells removes all *linear* decodability (linear leak-back of 0.00 in every cell) while the latent stays healthy by the rule of Section 3.1: effective rank stays at 77–99% of baseline and coordinate scale at 79–98% in every cell (absolute effective rank 11.4→9.9 at $\beta = 1$ and 5.4→4.6 at $\beta = 8$). A fresh *nonlinear* probe recovers the factor in every cell, but how much it recovers depends on the factor (Table 6, Figure 5). Over the 32 cells, nonlinear leak-back ranges from 0.54 to 1.00 (mean 0.84, median 0.86). For shape, x, and y it is 0.80–1.00 to two decimals (means 0.95, 0.87, and 0.85; the lowest of these 24 cells is a y cell at 0.798): the concept survives in nonlinear form even though no linear probe can find it. For scale it is 0.54–0.84 (mean 0.68): at $\beta = 1$, about 45% of the nonlinearly recoverable scale information is no longer recovered after erasure, and the cells below 0.80 in Figure 5b are the seven scale cells at 0.54–0.77 and that one y cell at 0.798. This is partial destruction at a healthy latent, and it is factor-specific; it reproduces across

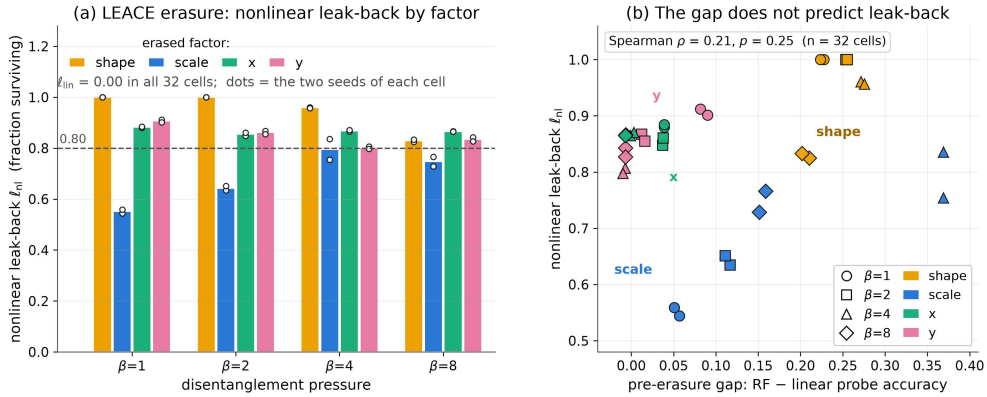


Fig. 5 LEACE concept erasure. (a) Nonlinear leak-back ℓ_{nl} by erased factor and β (bars: mean of two seeds; dots: the two seeds); linear leak-back ℓ_{lin} is 0.00 in all 32 cells by construction. Shape, x, and y stay at 0.80–1.00 to two decimals (lowest cell 0.798), whereas scale falls to 0.54–0.84: linear erasure mostly relocates the concept, but partially destroys scale. (b) The a-priori "nonlinear excess" gap (random-forest minus linear-probe accuracy on the original latent) does not predict ℓ_{nl} (Spearman $\rho = 0.21$, $p = 0.25$ over the 32 cells); points are colored by factor and shaped by β (legend in the panel), and the low group is scale.

all three VAE initializations (scale leak-back 0.51–0.58 at $\beta = 1$; Section 4.9) and lies far outside probe noise, since the two seeds of a cell differ by 0.015 on average and by at most 0.08. Averaged over factors, leak-back is flat across disentanglement levels (0.84/0.84/0.86/0.82 across $\beta \in \{1, 2, 4, 8\}$; Figure 5a), but these per-β means should not be read as a floor: the cell-level range is 0.54–1.00. Mean collateral damage to other factors is small (0.024), so the erasure is reasonably local.

Table 6 Nonlinear leak-back ℓ_{nl} after LEACE erasure of each factor, by disentanglement level (mean of two seeds), with the range and mean over all eight cells of that factor. Linear leak-back is 0.00 in every cell. Ranges are rounded to two decimals; the lowest y cell is 0.798. The shape entries of 1.00 at $\beta = 1$ and 2 are clipped values (Section 3.4); shape is also the one factor whose coarsened task is two-class with one removed dimension. Scale is the only factor whose cells fall well below 0.80; its per-β mean is below 0.80 at every β, although at $\beta = 4$ one of its two cells (0.84) lies above that line.

Erased factor	$\beta = 1$	$\beta = 2$	$\beta = 4$	$\beta = 8$	range (8 cells)	mean
shape	1.00	1.00	0.96	0.83	0.82–1.00	0.95
scale	0.55	0.64	0.79	0.75	0.54–0.84	0.68
x	0.88	0.85	0.87	0.87	0.85–0.88	0.87
y	0.91	0.86	0.80	0.84	0.80–0.91	0.85
all factors	0.84	0.84	0.86	0.82	0.54–1.00	0.84

4.8 Negative results for concept erasure

We tested two ways in which this could have become a stronger predictive or causal claim, and both came back negative. First, the a-priori predictor. A candidate pre-erasure predictor of nonlinear leak-back—the "nonlinear excess" gap, defined as the random-forest (RF) probe accuracy minus the linear-probe accuracy on the original latent—does not predict it (Spearman $\rho = 0.21$, $p = 0.25$; Figure 5b); x and y are near-perfectly linear (gap ≈ 0) yet leak back at ≈ 0.85–0.87, and scale has the lowest leak-back at every value of the gap. Second, disentanglement as a causal knob. Mean leak-back is flat across β (0.84/0.84/0.86/0.82), but the per-factor pattern is not: scale leak-back rises from 0.55 at $\beta = 1$ to 0.75–0.79 at $\beta \geq 4$ (Spearman $\rho = 0.83$, $p = 0.01$ over the eight scale cells, which come from four encoders) while shape falls from 1.00 to 0.83 ($\rho = -0.94$; Table 6). The negative result is therefore a null about the mean only: β does not make erasure less destructive overall, but it changes which factor is partially destroyed, and for scale, the one factor that is destroyed, raising β from 1 to 4 or more reduces the lost fraction from about 45% to 21–25%. We attribute the failure of the predictor to a structural fact: for three of the four factors the linear edit relocates rather than destroys, so the recoverability outcome saturates near 1.0 and leaves little variance to explain; the variance that does exist lies between factors, and scale is the exception.

4.9 Robustness across VAE initializations and disentanglement levels

To ensure the two relocation claims do not hinge on a single encoder, we retrain the β-VAE under three independent initializations (seeds 42, 7, 123) at $\beta = 1$ and re-measure both mechanisms; we additionally repeat Mechanism 1 across $\beta \in \{1, 2, 4\}$.

Across VAE initializations (three initializations, $\beta = 1$). Mechanism 1: permanent loss $1.1 \pm 0.6\%$, realignment recovery $90 \pm 6\%$, frozen-probe drop $27 \pm 2\%$. Mechanism 2: mean nonlinear leak-back 0.83 ± 0.01 with linear leak-back exactly 0.00. The factor-level pattern of Section 4.7 also reproduces: scale leak-back is 0.56, 0.51, and 0.58 in the three initializations, while shape is 1.00 in all three. Both relocation results are stable with small variance.

Across disentanglement levels (seed 42, $\beta \in \{1, 2, 4\}$). Relocation holds throughout (Table 7).

Recovery is not strictly monotone in β, but permanent loss stays small throughout: a fresh probe recovers the information to within 0.7–2.3 points at every β, and only the linear-map recovery dips. Relocation, rather than destruction, holds at every β for Mechanism 1, and the scale exception of Section 4.7 is reproduced here as well.

4.10 Summary of claims and evidence

Table 8 collects the claims the paper makes and the result that supports each. Every entry is stated at the strength the evidence supports, including the two caveats that qualify the relocation claim.

Table 7 Robustness of both relocation results across disentanglement levels (one VAE initialization, seed 42, per column; $\beta \in \{1, 2, 4\}$). Permanent loss stays small and mean nonlinear leak-back stays at 0.83–0.85 throughout, with scale alone at 0.56–0.78; only the linear-map recovery varies non-monotonically. The M2 rows are re-measured in the robustness run and therefore differ from the per-β means of Table 6 in the second decimal.

Metric	$\beta = 1$	$\beta = 2$	$\beta = 4$
M1 permanent loss	1.3%	2.3%	0.7%
M1 realign recovery	85%	72%	97%
M2 nonlinear leak-back (mean over factors)	0.84	0.83	0.85
M2 nonlinear leak-back (scale only)	0.56	0.63	0.78

Table 8 Each claim and the controlled experiment that supports it; the section references point to the relevant results.

Claim	**Supporting evidence**
Belief-revision drift mostly *relocates* information	Section 4.1: frozen probe breaks (19% at $b = 2$) while a fresh probe recovers (1.8% raw, 4.4% normalized loss); affine realign 85–90% across seeds (Tables 3 and 5). Loss rises with overlap and is carried by the scale factor
Relocation transfers to real CNNs; *linear* invertibility is substrate-dependent	Sections 4.2–4.3: affine recovery 96%/91% (MNIST) vs. 47%/13% (CIFAR); a fresh head recovers 0.974 vs. 0.990 (MNIST) and 0.748 vs. 0.712 (CIFAR)
Total loss arises through collapse, not drift (in our experiments)	Sections 4.4–4.5: recovery 87% to 0% and normalized loss 3% to 100% track rank/std collapse; drift correlates only modestly with loss within a sweep ($\rho = 0.45$–0.54) and fails across configurations ($\times$16: small drift, near-total loss); the $\times$4 row is a boundary case whose label rests on the scale criterion
Accumulated contradiction at a healthy step size does not destroy	Section 4.6: raw loss flat at 1–2% across 6→48 rounds
Linear concept erasure mostly relocates, with factor-dependent partial loss	Section 4.7: linear leak-back 0.00; nonlinear leak-back 0.80–1.00 for shape, x, y but 0.54–0.84 for scale, at a healthy latent (rank 77–99% of base)
No usable a-priori predictor; disentanglement redistributes leak-back across factors without changing its mean	Section 4.8: Spearman $\rho = 0.21$ ($p = 0.25$); mean leak-back 0.82–0.86 across β, while scale rises from 0.55 to 0.75–0.79 and shape falls from 1.00 to 0.83
Both relocation claims are robust to encoder init and β	Section 4.9: 3 VAE inits and $\beta \in \{1, 2, 4\}$; the scale exception reproduces in all three inits

5 Discussion

Both belief-revision drift (Sections 4.1–4.2) and LEACE concept erasure (Section 4.7) move factor information to new coordinates while leaving most of it recoverable. The learning-rate sweep (Sections 4.4–4.5) shows two regimes under the rule of Section 3.1: relocation (large change and a broken frozen readout, but high fresh recoverability and stable rank and std) and collapse (contracted variance and rank, fresh probes near chance, no recovery), with the $\times 4$ level sitting at the boundary between them. We had expected a third regime, a *healthy latent with irrecoverable information*, to be absent, and on the learning-rate sweep it is. The erasure experiment shows that it is not absent in general: when LEACE removes the scale factor, the latent stays healthy by every criterion we monitor (rank 82–93% of baseline), yet 16–46% of the nonlinearly recoverable scale information is gone. Partial destruction at a healthy latent therefore does occur, and it is factor-dependent. What we do not observe, on either mechanism, is *total* loss at a healthy latent: every configuration whose fresh probes fall to chance has also collapsed. We note that our grid makes this outcome hard to reach by construction: the belief-revision edit is confined to a two-dimensional subspace with a preservation penalty on the other factors' dimensions, and the erasure removes at most two of sixteen dimensions for one factor at a time, so the only operation on the grid that can push all four factors to chance is a diverging optimizer step, which also destroys variance (Section 6). The claim we draw from this is accordingly weaker than "relocation, never destruction": relocation is the dominant outcome of small linear edits on a healthy representation; partial destruction occurs, is factor-dependent, and is not predicted by change magnitude; and total loss, in our experiments, requires collapse. The two mechanisms are unified only at this level—both are linear post-hoc edits that mostly relocate under a healthy representation—and not mechanistically; we make no claim that belief revision and LEACE erasure are the same process.

Information recoverability is not the same thing as linear recoverability, and CIFAR-10 (Section 4.3) makes the distinction explicit: a frozen head can break, the factor can remain fully decodable (a fresh head recovers it and the features do not collapse), and yet a cheap affine map may undo only a small fraction of the drift. "The information survived" and "a linear map can restore the old readout" are different claims. Only the first held across all substrates in our experiments, and prior drift-compensation work should not assume that the second follows from it.

Change magnitude is a poor proxy for destruction. Within a single sweep it carries some information (Spearman $\rho = 0.54$ over the 15 overlap runs, where it tracks the small scale residue, and 0.45 over the 18 learning-rate runs), but it fails exactly where a proxy is needed. On dSprites, the learning-rate level with the smallest drift among the collapsed ones ($\times 16$, $\delta = 5.5$) has nearly total loss, while $\times 4$ and $\times 8$ drift three to four times as far and lose less; on the real encoders, the fine-tuning drifts ($\delta = 30.1$ on Split-MNIST and 10.4 on CIFAR-10, each in the units of its own feature layer and not comparable numerically with the dSprites latent) come with almost no loss. The rounds sweep likewise breaks the frozen head (mean frozen drop 22–27% at 6 rounds, 12–17% at 24, and 24–27% at 48) while the fresh information remains. Any monitoring scheme that equates representational displacement with forgetting will raise false alarms in relocation-dominant settings and can miss collapse when the displacement is modest.

The claim is not that representational edits are harmless. It is that under healthy latent conditions, frozen-readout failure is by itself insufficient evidence of information destruction, and that the amount actually destroyed must be measured with a fresh probe, factor by factor. Relocation also does not come with a usable a-priori predictor of how much leaks back, and disentanglement does not reduce erasure destructibility on average in our setting, although it shifts which factor is partially destroyed (Section 4.8). We flag both points so that the relocation result is not over-read as implying controllability or safety.

The result also carries a cautionary implication for erasure and unlearning practice. Because both belief-revision edits and linear concept erasure mostly relocate rather than remove factor information in our settings, an edit that appears to have erased or unlearned a concept may leave most of it recoverable by a nonlinear probe. Practitioners who use erasure or unlearning for privacy, safety, or fairness purposes should therefore not treat the failure of a linear readout as evidence of removal, and should verify removal with fresh, nonlinear recoverability tests. We make no claim that this behavior transfers to large models, where the stakes of such verification are higher.

For practitioners, the constructive implication is equally concrete: track latent variance and rank alongside task metrics, report them as ratios to the pre-edit baseline together with their absolute values, and when latent health is intact, a small affine realignment fitted on unlabeled post-edit latents may restore legacy heads without replay. Intact rank and scale are necessary, not sufficient: the scale cells of Section 4.7 pass both criteria while losing up to 46% of a factor, so partial destruction is visible only to a fresh, factor-wise probe. The thresholds we used (rank above 60% and scale above 40% of baseline) are heuristics that worked on our substrates; we offer them as a starting point, not as calibrated constants.

6 Limitations

Our results are on dSprites (β-VAE), Split-MNIST, and CIFAR-10 CNNs. We do not extend any claim to large models or language models, and we say nothing about cognition.

Statistical base. The headline relocation claims reproduce across three VAE initializations and $\beta \in \{1, 2, 4\}$ for Mechanism 1 and $\beta \in \{1, 2, 4, 8\}$ for Mechanism 2 (Sections 4.9 and 4.7); we do not run a larger initialization ensemble, and the collapse-frontier sweeps remain on the seed-42 encoder. The dSprites sweeps use five (overlap), three (learning-rate frontier), or two (collapse check, rounds, and each erasure cell) seeds; the belief-revision probes use a fixed random state and the erasure probes take the cell's seed; and the Split-MNIST and CIFAR-10 experiments are single training runs reported as point estimates. The high-pressure end of the learning-rate sweep, where normalized loss reaches 67–100%, is not in doubt, but orderings of small quantities at the low-pressure end (1.1% versus 1.5% permanent loss, 87% versus 83% recovery in Table 3) cannot be distinguished from run-to-run variation on this base, and at $\times 8$ and $\times 16$ one of the two seeds collapsed completely while the other did not.

Search for destruction at a healthy latent. The conclusion that total loss requires collapse is drawn from a grid that varies one factor, the revision learning rate, at one

overlap setting on one encoder, and the erasure grid coarsens every factor to three classes so that the removed subspace is at most two-dimensional. That coarsening is a control against trivial collapse, but it also excludes by construction the large-subspace edits most likely to destroy information at a healthy latent; the partial destruction of scale (Section 4.7) shows that the phenomenon exists even within this grid. An absence of total loss at a healthy latent on this grid therefore supports a weaker conclusion than it would on a grid designed to find it, and a sweep over larger erased subspaces (more classes, or several factors erased jointly) is the natural next experiment. Finer-grained concepts may behave differently.

Latent-health diagnostic. The health thresholds (rank above 60% and scale above 40% of baseline) are heuristics fixed in the analysis code, not calibrated quantities, and one row of the learning-rate sweep ($\times 4$) changes label under a rank-only rule (Section 4.5). Mean coordinate scale is not invariant to a uniform rescaling of the latent, so it is informative only in combination with rank and fresh decodability. The destroyed criterion is applied to the raw loss averaged over the four factors, so the total loss of a single 32-class factor (x or y, whose decodable range of about 0.32 contributes at most 0.08 to that mean) would not by itself cross the 0.10 threshold; no label in Tables 4 and 5 changes under the equivalent rule on the normalized loss (threshold 0.24, the raw threshold divided by the mean decodable range of 0.42), but the rule is insensitive to single-factor destruction. Absolute effective rank is reported for dSprites and CIFAR-10; the Split-MNIST run did not log it.

Scope. The collapse boundary is characterized through learning rate, and we do not exhaustively map task dissimilarity or width. The CIFAR-10 encoder is deliberately small (base accuracy 0.71), and we do *not* retune it to improve linear recovery, because the relocation finding does not depend on the affine map succeeding; a stronger encoder might well show different linear invertibility. Orientation is undecodable in our latents. Finally, we make no claim about cognitive plausibility or LLM safety, and none about universal continual-learning laws.

7 Conclusion

Our central argument is that the size of a representational change does not measure how much information was lost. We examined this on two distinct editing families, belief-revision drift and concept erasure, across three substrates (dSprites, Split-MNIST, and CIFAR-10). In both families relocation is the dominant outcome: the frozen readout breaks while a fresh probe recovers most of the factor. Relocation is not the whole story. Linear erasure partially destroys one factor (scale) while the latent stays healthy, and belief revision leaves a small, overlap-dependent residue of loss on the same factor, so partial destruction occurs and is factor-dependent. Total loss, in our experiments, arises only when the representation collapses under an excessive optimizer step size. A frozen head can fail and the drift can be large, yet neither fact—nor a weak linear recovery—implies that information has been lost, and a small drift does not imply that it has been kept. Accumulating contradictions does not change the picture when optimization stays healthy. Rather than treating representational change magnitude as a proxy for information loss, one should measure latent health and fresh,

factor-wise recoverability; when those are intact, a cheap affine realignment can serve as a first diagnostic, ahead of any appeal to replay or catastrophic forgetting. We present this as a characterization whose scope is bounded by explicit negative results and by the limitations stated above.

Supplementary information. The complete experiment code is provided as supplementary material in a single self-contained script (`reproduce_all.py`), together with the per-run `metrics.json` logs to which every reported number traces and the script (`reanalysis.py`) that derives the chance-normalized, per-factor, and per-seed quantities of this revision from those logs. The script's built-in documentation maps each experiment command to the table or figure it reproduces; datasets are downloaded automatically on first run. The same package is archived at Zenodo (see Code availability).

Declarations

Funding. No funding was received to assist with the preparation of this manuscript.

Competing interests. The author has no competing interests to declare that are relevant to the content of this article.

Declaration of generative AI use. Generative AI tools (large language models) were used to assist in the drafting and editing of this manuscript's text. All content was reviewed and verified by the author, who takes full responsibility for the integrity, accuracy, and originality of the manuscript.

Ethics approval and consent to participate. Not applicable. This work uses only publicly available image benchmarks (dSprites, MNIST, CIFAR-10) and involves no human participants or animals.

Consent for publication. Not applicable.

Data availability. All datasets used in this study are publicly available: dSprites [2] from the DeepMind repository, and MNIST and CIFAR-10 via `torchvision`. The per-run `metrics.json` logs underlying every table and figure are included in the supplementary material and archived at Zenodo under https://doi.org/10.5281/zenodo.23914655.

Code availability. The complete experiment code (`reproduce_all.py`, `reanalysis.py`, the figure script, and the per-run `metrics.json` logs) is archived at Zenodo under https://doi.org/10.5281/zenodo.23914655 and is also provided as supplementary material.

Author contributions. S.H. designed the study, implemented the experiments, analyzed the results, and wrote the manuscript.

Appendix A Implementation details

β-VAE.

All dSprites experiments share one β-VAE. The encoder is four 4×4 stride-2 convolutions ($1 \to 32 \to 64 \to 128 \to 256$ channels with ReLU between), flattened to 4096 and mapped by two linear heads to a 16-dimensional posterior mean and log-variance; the decoder mirrors this with four transposed convolutions and a sigmoid output, and we take latents to be the posterior mean. Training uses Adam (learning rate 10^{-3}), 25 epochs, batch size 128, and loss $\mathrm{BCE}(\hat{x}, x)/\mathrm{batch} + \beta\,\mathrm{KL}$ with $\beta = 1$ unless stated otherwise (the disentanglement robustness arm uses $\beta \in \{1, 2, 4\}$ and the erasure sweep $\beta \in \{1, 2, 4, 8\}$). Inputs are the binary dSprites images kept in $\{0, 1\}$ (not divided by 255, which collapses the posterior); we use 200k training and 10k evaluation images and global seed 42.

Belief module and Mechanism 1.

Beliefs are shape-to-color associations carried by a small vector-symbolic module (shape and color prototypes in $\mathbb{R}^{16}$, a $\mathrm{Linear}(16, 64) \to \mathrm{ReLU} \to \mathrm{Linear}(64, 3)$ readout, with circular-convolution binding); initial beliefs are trained for 15 epochs (Adam, 5×10^{-4}). Each revision round applies one contradiction for 10 epochs of projection-based revision (Adam, base learning rate 5×10^{-4}): the decoder is frozen and the gradient is confined to a 2-dimensional belief subspace, with a preservation penalty (weight 10) on the 8-dimensional subspace Ω spanned by the two latent dimensions assigned to each of the other four factors (scale, orientation, x, y). The overlap sweep sets the number of true semantic dimensions inside the belief subspace to $b \in \{0, 1, 2\}$; the learning-rate sweep multiplies the revision learning rate by $\{1, 2, 4, 8, 16, 32\}$ (frontier) or $\{1, 4, 8, 16, 32\}$ (collapse check, a separate run that also logs latent health); the rounds sweep runs 6, 24, and 48 rounds at the base rate. The number of partition seeds per configuration is five for the overlap sweep, three for the frontier sweep, and two for the collapse check and the rounds sweep; Tables 3 and 4 therefore come from different runs, and their loss values differ slightly at $\times 8$ and $\times 16$.

Mechanism 2 (LEACE).

We apply the closed-form LEACE eraser of Section 3.1 to the 16-dimensional latent, coarsening each factor to $K = 3$ quantile classes (two-class in effect for shape, Section 3.4), and sweep $\beta \in \{1, 2, 4, 8\}$ across the four decodable factors with two seeds per cell (32 cells in total); each seed fixes the 50/50 split, the eraser fitted on its training half, and the probes' random state.

Probes and realignment.

Decodability is measured per factor on a 50/50 split of the evaluation latents: a frozen probe is trained on pre-edit latents and scored on post-edit latents, while a fresh probe is retrained from scratch on post-edit latents. Nonlinear probes are random forests (100 trees in the overlap sweep, 120 in the learning-rate, collapse, and rounds sweeps, and 150 in the erasure and robustness runs; the random state is fixed except for the erasure cells, which use the cell's seed); linear probes and the retrained upper

bound are logistic regression. The affine realignment map is the least-squares solution of Section 3.1 (the replay-free variant is fit on current-task latents only), and the orthogonal variant is the corresponding Procrustes solution.

Latent-health metrics.

Effective rank is the participation ratio of the latent-covariance eigenvalues, $\mathrm{Tr}(\Sigma)^2/\mathrm{Tr}(\Sigma^2) \in [1, d]$; mean scale is the per-dimension standard deviation averaged over dimensions; drift is the mean row-wise $\|z^{\mathrm{a}} - z^{\mathrm{b}}\|_2$. Rank and scale are reported as ratios to their pre-edit baseline, with absolute values given in the text (dSprites: rank 11.4 of 16, scale 0.73; CIFAR-10 feature layer: rank 2.69 of 128, scale 0.44, before the edit). The regime labels of Tables 4 and 5 use the rule of Section 3.1: healthy if the rank ratio exceeds 0.6 and the scale ratio exceeds 0.4; destroyed if the raw permanent loss averaged over factors exceeds 0.10.

Real-encoder substrates.

For *Split-MNIST*, the encoder is three 3×3 stride-2 convolutions ($1 \to 32 \to 64 \to 128$, ReLU) followed by global average pooling and a Linear$(128, 128)$ feature layer, with linear heads for Task A (digits 0–4) and Task B (digits 5–9); we train head A on Task A, then fine-tune the full encoder with a fresh head B on Task B (Adam, 10^{-3}, 5 epochs each). For *CIFAR-10*, the encoder is three convolutional blocks ($3 \to 32 \to 64 \to 128$, 3×3, batch normalization, ReLU, max-pooling) with a 128-dimensional feature layer; the split is classes 0–4 versus 5–9, trained for 15 then 8 epochs (Adam, 10^{-3}) on standard-normalized inputs. Both real-encoder runs use a single seed, and the CIFAR encoder is deliberately small and is not retuned.

Seeds, compute, and software.

The shared β-VAE is trained at global seed 42; the robustness study additionally retrains it at initialization seeds $\{42, 7, 123\}$. Experiments run on a single GPU/MPS device in PyTorch with scikit-learn probes; trained encoders are cached and reused across sweeps, and all reported numbers trace to per-experiment `metrics.json` logs. All figures were produced with Matplotlib. We additionally document a normalization pitfall (the 1/255 posterior-collapse issue of Section 3.2) so that the decodability baselines are reproducible.

References

[1] Higgins, I. *et al. β-VAE: Learning basic visual concepts with a constrained variational framework. International Conference on Learning Representations (ICLR)* (2017).

[2] Matthey, L., Higgins, I., Hassabis, D. & Lerchner, A. dSprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset/ (2017).

[3] Belrose, N. *et al. LEACE: Perfect linear concept erasure in closed form. Advances in Neural Information Processing Systems (NeurIPS)* (2023).

[4] Li, Z. & Hoiem, D. *Learning without forgetting. European Conference on Computer Vision (ECCV)* (2016).

[5] Rebuffi, S.-A., Kolesnikov, A., Sperl, G. & Lampert, C. H. *iCaRL: Incremental classifier and representation learning. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

[6] Yu, L. *et al. Semantic drift compensation for class-incremental learning. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020).

[7] Gu, Q., Shim, D. & Shkurti, F. *Preserving linear separability in continual learning by backward feature projection. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023).

[8] Gomez-Villa, A. *et al. Exemplar-free continual representation learning via learnable drift compensation. European Conference on Computer Vision (ECCV)* (2024).

[9] Davari, M., Asadi, N., Mudur, S., Aljundi, R. & Belilovsky, E. *Probing representation forgetting in supervised and unsupervised continual learning. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022).

[10] Ravfogel, S., Elazar, Y., Gonen, H., Twiton, M. & Goldberg, Y. *Null it out: Guarding protected attributes by iterative nullspace projection. Annual Meeting of the Association for Computational Linguistics (ACL)* (2020).

[11] Ravfogel, S., Twiton, M., Goldberg, Y. & Cotterell, R. *Linear adversarial concept erasure. International Conference on Machine Learning (ICML)* (2022).

[12] Ravfogel, S., Vargas, F., Goldberg, Y. & Cotterell, R. *Adversarial concept erasure in kernel space. Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2022).

[13] Liu, P. & Zhang, C. Erased or dormant? Rethinking concept erasure through reversibility. arXiv preprint arXiv:2505.16174 (2025).

[14] Mendelson, M. & Belinkov, Y. *Debiasing methods in natural language understanding make bias more accessible. Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2021).

[15] Xu, X. *et al.* Unlearning isn't deletion: Investigating reversibility of machine unlearning in LLMs. arXiv preprint arXiv:2505.16831 (2025).

[16] Dohare, S. *et al.* Loss of plasticity in deep continual learning. *Nature* **632**, 768–774 (2024).

[17] De Lange, M., van de Ven, G. M. & Tuytelaars, T. *Continual evaluation for lifelong learning: Identifying the stability gap. International Conference on Learning Representations (ICLR)* (2023).

[18] Alain, G. & Bengio, Y. Understanding intermediate layers using linear classifier probes. arXiv preprint arXiv:1610.01644 (2016).

[19] Hewitt, J. & Liang, P. *Designing and interpreting probes with control tasks. Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2019).

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- supplementarycode.zip