Experiment 2.9.2: fallback control for deleting *red*

> We compared two remedies for ablation seed-sensitivity: **1.** optimal ablation, which picks a replacement constant after training, and **2.** fallback control, a decoder-only loss term that teaches the model what to output once the concept has been removed. Fallback control worked well. Neither remedy touches the other half of the variance, where anchoring itself fails.

Ex-2.9.1 reproduced the main M1 result: anchor *red* to the first latent axis, zero the axis, and the reconstruction error concentrates on red-like colors. It also reproduced the main weakness: the outcome varies a lot with the random seed. The original study handled that with a 60-seed sweep and Pareto selection, but that won't scale: as models grow, the chance that any given seed allows clean intervention may shrink. So this experiment tries a different approach.

The SCA paper suggests optimal ablation (Li & Janson 2024) as a fix: replace the removed component with an optimized constant. We test that and also a training-time alternative we'll call "fallback control". The plan is to teach the decoder, during training, what to output once the concept has been removed.

Why weight ablation is noisy

The bottleneck is unit-normalized, so zeroing an encoder axis renormalizes whatever is left. For a red-like input the remainder is small and close to random, just the residual geometry that this particular seed happened to produce, so ablated red is decoded as some other arbitrary color. Li & Janson call this "spoofing": the intervention removes the concept and inserts a random claim about the input. Two seeds with identical anchoring quality can then score very differently, purely on where red happens to land.

Optimal ablation measures importance while minimizing spoofing. It replaces the removed component with the constant $a^* = \arg\min_a \mathbb{E}[\mathcal{L}]$, the value that affects loss the least. But for concept removal, that objective points the wrong way: the loss it minimizes includes the loss on the target itself, so a^* gets pulled toward whatever constant best restores *red*. We evaluate both the literal method (`oa`) and a removal-appropriate version (`oa-nontarget`, which optimizes the constant over non-red colors only).

Fallback control instead makes the post-removal behavior a trained property. The anti-anchor regularizer already keeps the direction $-e_1$ empty, so we add one decoder-only loss term, MSE(dec($-e_1$), mid-gray). This pins that reserved direction to a *null* output of our choosing. A model that has genuinely lost the color information should fall back to a noncommittal guess, and the least committal guess over the RGB cube is mid-gray, so we pick mid-gray as *null*. An intervention can then redirect red to $-e_1$ and get a predictable response.

Conditions

We run two training variants, otherwise identical to ex-2.9.1 (same model, data, dopesheet), with 32 seeds each: `base` (the ex-2.9.1 loss, unchanged) and `fallback` (the same loss plus 0.05 × the fallback term). Each trained model is then scored under five weight-level interventions on the first axis:

- `zero` — zero the encoder's first output row and its bias (the status quo).
- `oa` — zero that row, then set the bias to the constant that minimizes mean reconstruction error over the full grid (optimal ablation, taken literally).
- `oa-nontarget` — the same as `oa`, but the constant is optimized over non-red colors only.
- `reflect` — negate that row and its bias, so $z_1 \to -z_1$ and red lands exactly on $-e_1$. This is a clean redirect, though not a true deletion, since a sign flip undoes it.
- `redirect` — zero that row and set the bias to −1. The redness computation is deleted, and inputs are also nudged toward $-e_1$ in proportion to how much of their pre-norm activation the deletion removed. This is permanent, like `zero`, but with a defined destination.

Each run is scored like M1 Ex-2.9.1: the R^2 between per-color reconstruction error after the intervention and (HSV similarity to red)^3. The experiment lives in `experiment.py`:

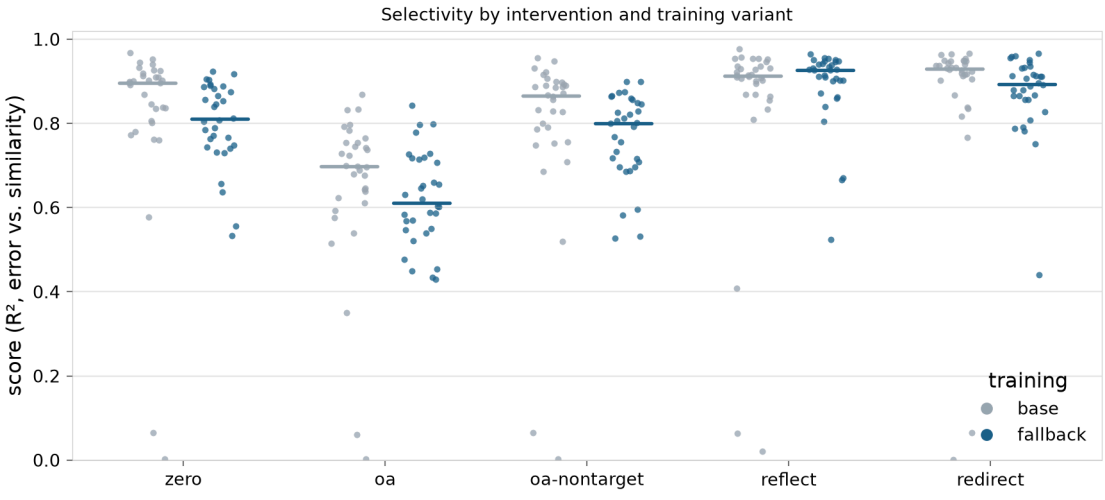
```
bin/mini run docs/m1/ex-2.9.2/experiment.py --app moda
```

64 runs completed (32 seeds × 2 variants).

On the selectivity, `base + zero` gets 0.82 ± 0.22 (min 0.00) and `fallback + reflect` gets 0.89 ± 0.10 (min 0.52). That looks like an improvement, but most of the gap comes from two base seeds that failed badly. The clear change is in the response. Damage to pure red goes from 0.31 ± 0.15 across seeds to 0.231 ± 0.020, pinned just under the analytic ¼ bound.

Selectivity across seeds

What matters here is what happens to the bad seeds, because the floor and spread of the distribution decide how many seeds you need to run. Each dot below is one trained model, and the bar is the median of the condition.

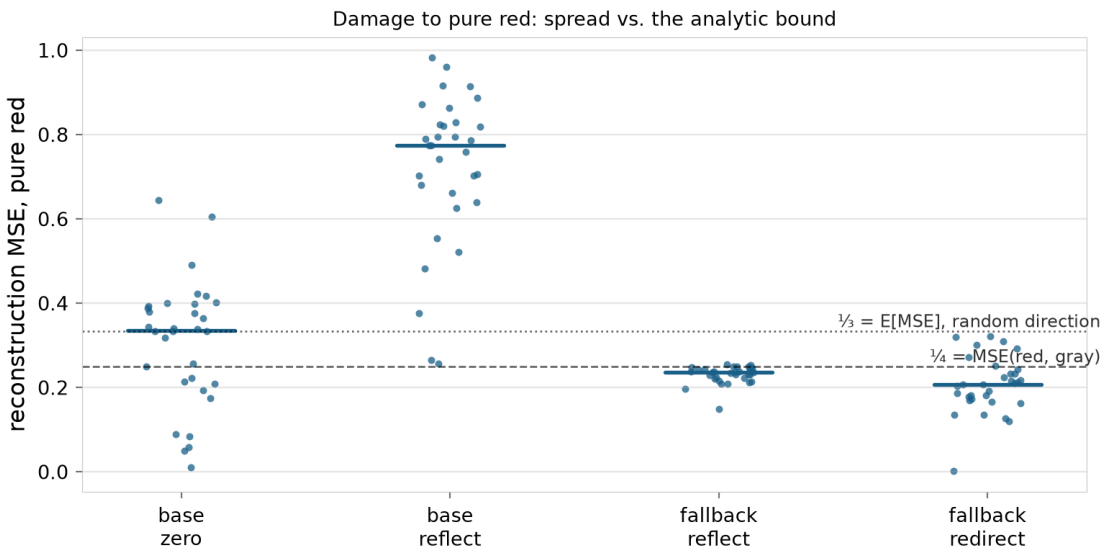


The scatter shows two things. First, the `base` variant has 2 seeds scoring below 0.1 under every intervention. Those are anchoring failures (in the worst one, red never anchored at all: a validation anchor loss of 0.61, against a median of 0.006), and no intervention-time trick reaches them. The fallback variant happened to produce none of these in 32 seeds. That is encouraging, but training is chaotic enough that we can't separate one extra loss term from plain luck. Second, among the seeds that did anchor, the R^2 distributions barely move: `base + zero` excluding failures scores 0.87 ± 0.08, against 0.89 ± 0.10 for `fallback + reflect` and 0.87 ± 0.10 for `redirect`. By this metric, fallback control adds little.

R^2 measures whether error is proportional to redness, but it says nothing about whether the size of the response is the same from seed to seed; we'll look into that below. (One aside: plain zeroing scores a little lower on the fallback variant, median 0.81 vs 0.90. The fallback term presumably perturbs the decoder that zero-ablated latents still pass through, so a trained fallback should be paired with its redirect rather than zeroing.)

A predictable response

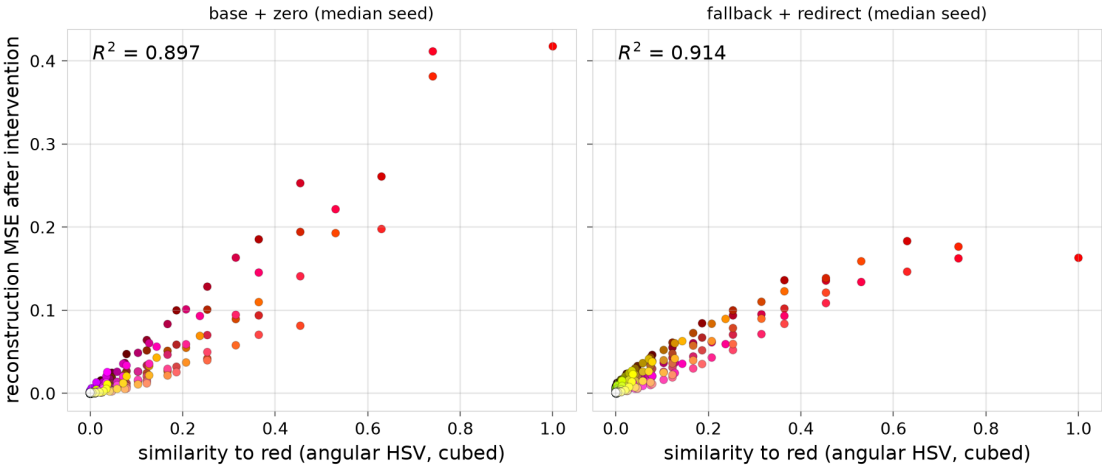
SCA aims for bounded side effects. With the decoder pinned to mid-gray at $-e_1$, redirecting pure red there should give MSE(red, gray) = ¼. Without fallback training there is no bound at all, and red lands wherever the untrained region happens to decode. The expectation for zero ablation under random redistribution is ⅓, but the seed-to-seed spread makes that expectation unhelpful.

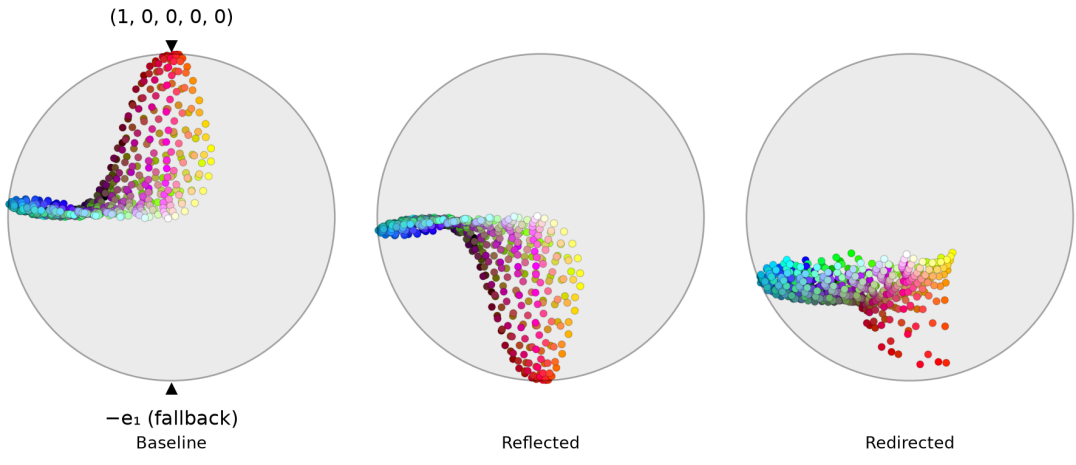


Under `base + zero`, damage to pure red spans 0.01–0.64 across seeds. On some seeds the "deleted" red reconstructs almost perfectly, because renormalization handed it to a neighboring color. `base + reflect` produces large damage (0.72 ± 0.18), but with no control over where red ends up. `fallback + reflect` lands at 0.231 ± 0.020, just under the ¼ bound predicted from the geometry, while collateral damage on non-red colors stays at 0.0010.

Median proportionality

The charts below show the median-scoring runs of each variant.





The mechanism shows up in the geometry. Both redirect-style interventions move red to $-e_1$, a region the anti-anchor regularizer kept empty, and fallback training decided what lives there. Across the 32 fallback runs, dec($-e_1$) is (0.50, 0.50, 0.50), with a maximum per-channel deviation of 0.004: mid-gray on every seed. In the base variant the same point decodes to an uncontrolled color (per-channel spread 0.21), which is why `base + reflect` damages red heavily but unpredictably.

Why optimal ablation scores lower here

Optimal ablation scored lower than plain zeroing on selectivity: the OA constant minimizes expected loss over the task distribution, and that distribution includes the concept being removed. In an anchored model the cheapest way to reduce expected loss is to partially restore the concept and shrink the error signal that the score measures. This makes sense for the question OA was built for, where a small loss gap is the useful outcome.

The removal-appropriate version, which optimizes the constant over non-red colors only, lands close to zero. The SCA anti-subspace regularizer already made small constants the least disruptive for bystander colors during training, so this version behaves like plain zeroing. Anchoring, in other words, gives you most of the "optimal" part of optimal ablation for free.

The literal OA constant is positive (red-ward) in 63 of 64 runs (median +0.50, except for the anchoring failures). With OA, the damage to pure red drops to 0.091 on the base variant, against 0.307 for zeroing, so the deletion is partially undone.

Takeaways

The seed problem from ex-2.9.1 seems to be two problems. One is intervention unpredictability, and in this experiment that was solved by fallback control: one decoder-only loss term gives the intervention a designed outcome, bounded above by ¼ for pure red and within 0.02 of that bound across every seed. The other is anchoring failure, where the concept never isolates onto its axis, and nothing applied at intervention time can fix that.